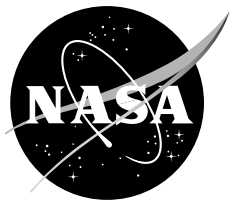
NASA/CP–2007-213468



Intelligent Motion and Interaction Within Virtual Environments

Editors:

Stephen R. Ellis
Ames Research Center
Moffett Field, California, U.S.A.

Mel Slater
University College, London, U.K.

Thomas Alexander
FGAN-FKIE
Wachtberg-Werthoven, Germany

Proceedings of a workshop sponsored
by the National Aeronautics and Space
Administration and held at
University College,
London, U.K.
15-17 September 2003

June 2007

The NASA STI Program Office . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) Program Office plays a key part in helping NASA maintain this important role.

The NASA STI Program Office is operated by Langley Research Center, the Lead Center for NASA's scientific and technical information. The NASA STI Program Office provides access to the NASA STI Database, the largest collection of aeronautical and space science STI in the world. The Program Office is also NASA's institutional mechanism for disseminating the results of its research and development activities. These results are published by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA's counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.

- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.

- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or cosponsored by NASA.

- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services that complement the STI Program Office's diverse offerings include creating custom thesauri, building customized databases, organizing and publishing research results . . . even providing videos.

For more information about the NASA STI Program Office, see the following:

- Access the NASA STI Program Home Page at *http://www.sti.nasa.gov*

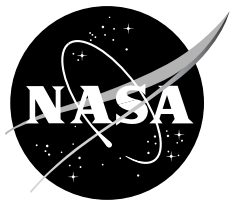
- E-mail your question via the Internet to help@sti.nasa.gov

- Fax your question to the NASA Access Help Desk at (301) 621-0134

- Telephone the NASA Access Help Desk at (301) 621-0390

- Write to:
  NASA Access Help Desk
  NASA Center for AeroSpace Information
  7115 Standard Drive
  Hanover, MD 21076-1320

NASA/CP–2007-213468



Intelligent Motion and Interaction Within Virtual Environments

Editors:

Stephen R. Ellis
Ames Research Center
Moffett Field, California, U.S.A.

Mel Slater
University College, London, U.K.

Thomas Alexander
FGAN-FKIE
Wachtberg-Werthoven, Germany

Proceedings of a workshop sponsored
by the National Aeronautics and Space
Administration and held at
University College,
London, U.K.
15-17 September 2003

National Aeronautics and
Space Administration

Ames Research Center
Moffett Field, California 94035-1000

June 2007

Acknowledgments

The authors would like to acknowledge the role of Dr. Terry Allard, Division Chief for the Human Factors Research and Technology Division at Ames Research Center, who provided the original inspiration for this conference and arranged for the required fiscal support, mainly from Code UL at NASA Headquarters. The original idea for this conference arose from discussions between Dr. Allard, Dr. Steven Zornetzer, and other members of the NATO IST-029/RTG-011 group. Lissa Webbon of Ames Research Center aided in the preparation of this document, and J. J. Giwa of University College, London, was instrumental in making local arrangements for the conference.

Available from:

NASA Center for AeroSpace Information
7115 Standard Drive
Hanover, MD 21076-1320
(301) 621-0390

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22161
(703) 487-4650

This report is also available electronically at:
http://http://human-factors.arc.nasa.gov/web/library/publications/publications.php

Table of Contents

Conference Proceedings:
Intelligent Motion and Interaction Within Virtual Elements
University College, London, United Kingdom
15–17 September 2003

Introduction

Stephen R. Ellis
Ames Research Center
Moffett Field, California 94035-1000
U.S.A.

What makes virtual actors and objects in virtual environments seem real? How can the illusion of their reality be supported? What sorts of training or user-interface applications benefit from realistic user-environment interactions? These are some of the central questions that designers of virtual environments face. To be sure simulation realism is not necessarily the major, or even a required goal, of a virtual environment intended to communicate specific information. But for some applications in entertainment, marketing, or aspects of vehicle simulation training, realism is essential. The following chapters will examine how a sense of truly interacting with dynamic, intelligent agents may arise in users of virtual environments. These chapters are based on presentations at the London conference on Intelligent Motion and Interaction within Virtual Environments which was held at University College, London, 15–17 September 2003. The full set of presentations at the conference may be found on the Web at:

http://www.cs.ucl.ac.uk/Motion/

The meeting was sponsored by the Human Factors Research and Technology Division of the NASA Ames Research Center, Moffett Field, CA; the University College, London, U.K.; and Electronic Arts, Canada.

Constraints

The organizing hypothesis for the conference, as well as the following set of papers, was that a sense of intelligent interaction with agents would more easily arise when simpler lower level aspects of the agents, such as their kinematic and dynamic description, are simulated with high fidelity. The effect is thought to occur in the same way that intelligent behavior may seem to emerge from carefully built low level modules in a subsumption architecture (Brooks, 1986). In the robotics case, the higher fidelity, low level behavior generation simplifies the higher level task simulation by exclusion from consideration or emergent behaviors that would be inconsistent with the lower level modeling.

Figure 1 illustrates several levels at which this simplification could take place as constraints from lower levels propagate upwards. The least restricted movement would be that of physical motion of freely moving impenetrable objects constrained only by Newton's three laws, friction, and collisions with each other or with fixed structures. The mechanical linkages of limbs provide additional constraints on motion that can give it the appearance of natural biological motion such as walking without invoking high level control. Higher level goal seeking or obstacle avoiding processes would provide the most restrictive strictures, but the proposal is that creation of such motion should be easier and more creditable if the lower level constraints are first satisfied.

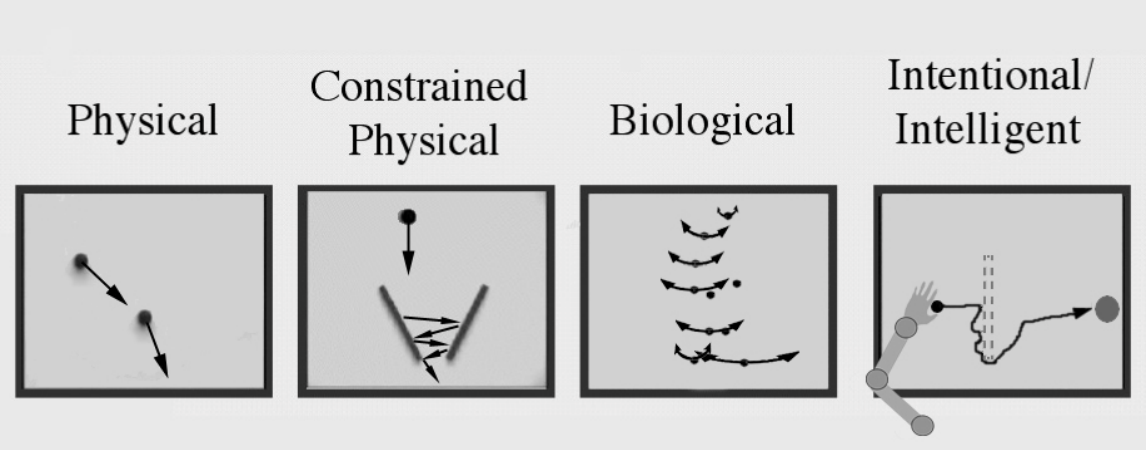


Figure 1. The increasing extent of constraint on motion as consideration moves from a idealized physical interaction between colliding or deflected balls (far left and center left), through biological motion of a walking person, to an intentionally controlled cursor subject to the kinematic, dynamic, and cybernetic constraints of arm movement. In this case the controlled elements is made to avoid an invisible barrier in order to hit a target on the other side (far right). The arrows approximate the patterned internal motion within each panel.

Interaction

Though the sense of inclusion in a synthetic or remote space may be the most striking immediate feature of a virtual environment, its most significant attribute with respect to use in training or as a user-interface is that it is interactive. A user in a virtual environment may move around within it and, provided that software support exists, interact with virtual objects or actors that are included. The consequent movement present within the environment may take on various forms (Figure 2). 1) It may be free running, independent of user action except for user observation, or it may be contingent on user action. This could correspond to a dimension called *contingency* of interaction. 2) It could involve user self-motion or motion of external entities along a dimension that may be called *otherness*. And, 3) it may or may not involve articulated action; that is motion which takes place with respect to another potentially moving element. This dimension could be called *articulation* with its value varying with the degree of complexity of the underlying articulated linkage.

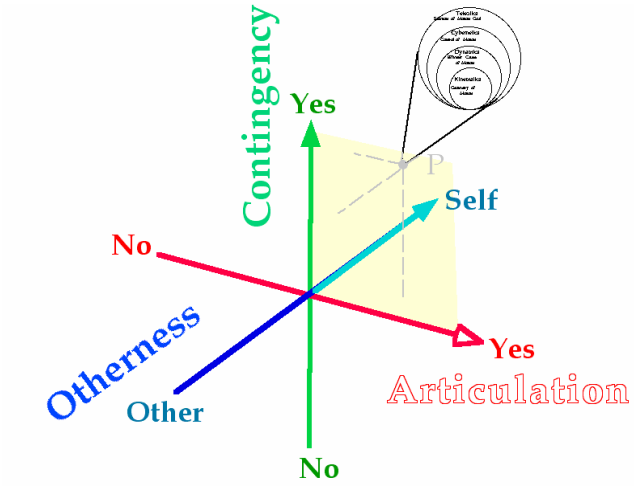


Figure 2. Some dimensions in which motion may be classified.

These three characteristics provide some dimensions by which motion in a virtual environment may be classified. They may be incorporated into an explanatory hierarchy (Figure 3) in which increasingly more abstract causes are used as explanations for the appearance of increasingly high level interaction.

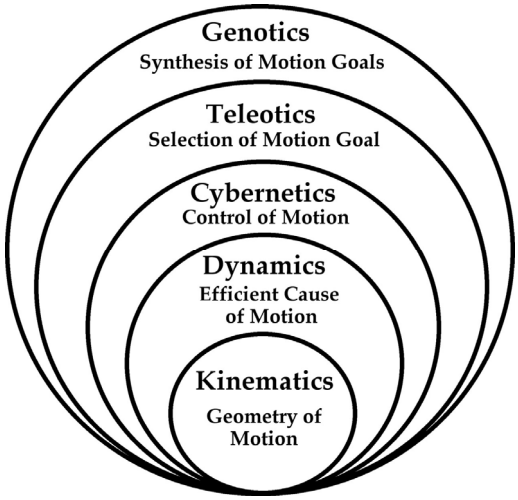


Figure 3. A hierarchical analysis of the causal influences on motion.[1]

[1] This classification of causes of motion can be related to Aristotle's classification of causes. The geometry of the kinematic level correspond to the material cause. The forces and torques of the dynamics correspond to the efficient cause. The cybernetic goals correspond to the final causes. And the abstractions of the teleotic level which govern goal selection correspond to the formal causes. Aristotle's original classification is, however, extended by the genotic level since it envisions the creation of forms not preexisting.

As discussed below, this hierarchy of causality rises from kinematic, dynamic, and cybernetic explanations to explanations that may be neologistically described as teleotic or genotic. The first three levels are used in the same sense as in conventional control theory. Kinematic explanations are associated with the purely geometric aspect of motion constraints such as link lengths, joint properties, and coupling characteristics. Dynamic explanation builds on underlying geometry but introduces forces and torques as causes for changes of motion. The cybernetic level introduces the ideas of a goal, error, noise, and a control law used to approach the goal. The teleologic and genologic levels are less conventional terms but naturally extend the hierarchy. A teleologic level of explanation involves rules for selecting goals, not merely the seeking of a goal. The highest level explanation, the genologic, which is required for generalized intelligent interaction, involves systems for the generation of new goals.

A wide variety of approaches have been attempted with respect to simulation of intelligent interaction, and there has been a significant criticism of purely algorithmic approaches, especially for genologic interaction. The extremely large variety of possible goals certainly provides major obstacles to the algorithmic approach. But the wide variety of goals and the motion necessary to reach them are also subject to an equally extensive variety of constraints arising from the physical facts associated with any particular system.

This observation then leads to a hypothesis that we may adopt as an organizing principle for framing the following chapters: the genesis of apparently intelligent interaction arises from an upwelling of constraints determined by a hierarchy of lower levels of behavioral interaction. In this sense intelligent interaction is a feature that emerges from an upward cascade of constraints in the same way that intelligent robot behavior is thought to arise from R. Brooks' subsumption architecture.

An example of how upwelling constraints simplify explanation or modeling at a higher level is specifically found in the relationship between kinematic and dynamic models. The geometric description of the linkage of a robotic arm, for example, highly constrains the possible motion that can be produced by a wide variety of forces. Thus, any correct explanation or model of the consequences of applied force that is an explanation at the dynamic level is directly determinable if the lower level structure is well known. Clearly, the crux of this relationship is the ability of the modeler to represent the facts of the lower level of analysis in terms of the higher level. This difficulty is readily solved in the case of dynamics since the explanatory concepts of force and torque used are defined with respect to geometric terms of the just lower level. Thus, what we should seek in general are analogous definitions of explanatory concepts between adjacent levels farther up in the hierarchy. That is to say, cybernetic control laws should be expressed in the language of dynamics, teleotic principles of goal selection should be expressed in the language of cybernetic control, and genotic syntheses of possible goals should be expressed in terms of the teleotic selection descriptions. It is important to note that the subsumption of each level within all higher levels, as indicated in Figure 3, means that higher levels have access to all lower level motion information so that the analysis does not represent a true hierarchy.

Papers

Ten papers and reviews based on presentations at the London meeting were submitted and are discussed below. They touch on aspects of the basic theme in a variety of ways.

Thomas Sheridan's Chapter 1, *Constraint, Intelligence, and Control Hierarchy in Virtual Environments,* provides examples of the role of constraints in defining intelligent behavior. The contrast between music and noise, for example, may seem to be based on musical sounds satisfying a number of constraints originating within the human nervous system. He also discusses how intelligence may be attributed to agents whose behavior satisfied contextually determined expectations. He stresses the hierarchical nature of the control effected by multiple constraints and the role of the optimization techniques in generating intelligent behavior.

Using the concrete and detailed example provided by a high performance hand gesture video recognition system, Karl-Friedrich Kraiss and Jorg Zieren explicitly show in Chapter 2, *Hierarchically Structured Non-Intrusive Sign Language Recognition,* how lower level kinematic and dynamic constraints that are introduced in the form of Kalman filters, aid automatic recognition of sign language gestures. The system they describe is among the best in the world for their task.

In contrast to Kraiss and Zieren who provide an example of intelligence on the part of an individual agent, Ronald Kruk provides concrete examples of more social activity in Chapter 3, *Intelligent Entity Behavior Within Synthetic Environments*. He considers what it takes to make groups of automatically controlled threat aircraft convince users of flight simulators that they are, in fact, under attack by enemy aircraft adhering to known military doctrines. This contribution emphasizes the importance of detailed and extensive physical modeling for creating a visually realistic virtual environment for flight simulation. The need for a hierarchical simulation structure is also emphasized.

Cardullo, Lewis, and Pafilov's Chapter 4, *Telerobotic Surgery: An Intelligent Systems Approach to Mitigate the Adverse Effects of Communication Delay,* provides an example of a solution to a problem that plagues virtual environment and teleoperation systems alike: transport delay. In virtual environments it can be caused primarily by rendering delays, whereas in teleoperation systems it arises from communication delays. They explore the mitigation technique of a temporal "clutch" which can be introduced to a simulator or teleoperator system as a predictor display. Such predictors need to be carefully driven to produce movements that appear intelligently controlled. The predicted movements on these displays, which are interpolated into the simulation, require the prediction systems to develop accurate computational models of the user. These models are likely to be most effective if they can be hierarchically built on kinematic, dynamic, cybernetic, and cognitive levels. Such models which can anticipate user motion in latency environment must incorporate intelligence both in sensory processing and separate in motor processing. Thus, optimal interaction with a user-interface for teleoperation can require modeling of intelligent behavior within the interface itself.

The high dynamic fidelity that Cardullo et al. seek to provide to teleoperator users arises partly from highly realistic visual content originating from a camera. Similarly, detailed content can also be provided by environmental simulations, but the computational cost is high. Nebel, Sibirjakov, and Ju in Chapter 5, *V-Man Generation for 3-D Real Time Animation,* illustrate some of these costs by providing a detailed example of the geometric, kinematic, and dynamic modeling necessary to produce detailed, embodied, humanoid agents in an interactive, physical simulation. Many approaching high level applications in virtual environment have in the past not appreciated the tremendous amount of notational, mathematical, and computational complexity that is required for creation of a flexible and expressive virtual environment. Careful attention to this paper will quickly remedy such an error. In fact, this paper merely touches on some of the content definition issues and only alludes to the equally complicated user, object, and avatar interaction that also must be determined. In fact, much of the art of virtual environment synthesis is to understand all the environmental options that may be voluntarily or inadvertently controlled by the designer. This chapter will help with the understandings of these specifics.

In a more general discussion of virtual environtments as a system, Slater and Sanchez-Vives examine in Chapter 6, *Interactions with Virtual People: Do Avatars Dream of Digital Sheep?,* what is it that prompts physical users to interact with virtual entities as if they were real agents. They are also interested in the parallel question about what in general prompts users to attribute reality to virtual objects. In the search for answers to these questions, they explore whether one can find indicators of the users' sense of personal presence and the presence of the avatars and virtual objects. They are particularly amazed by the users' tendency to reify the virtual and react emotionally to what must at some level be considered merely phantoms of light, reflections in Plato's Cave, as it were. They discuss some potential explanations from a neuroscience point of view. They conclude with an observation that interaction with virtual agents will become more and more common in daily life as business and industry develop virtual agents for the new electronic media, i.e., the World Wide Web, Wi-Fi enabled Personal Digital Assistants, Information Kiosks/Automated Tellers, and the now ubiquitous mobile phones.

Johnson in Chapter 7, *Dramatic Expression in Opera, and Its Implications for Conversational Agents,* also addresses the operation of a virtual environment as a whole. He examines a variety of operatic techniques such as the role of exaggeration in communication. He draws on the dramatic and rhetorical techniques used in staging opera to suggest principles to be used in the design of intelligent interaction within virtual environments. One goal of such techniques would be to assure effective communication in the presence of noise and distraction. In this respect his proposal is analogous to the techniques of filtering and exaggeration used in medical illustration, which produce images more realistic than the reality they reproduce. He also considers how an understanding of the causal structure of a story and the consistency of characters' behavior aids in the overall believability of their action. Finally, his discussion provides another example of hierarchical control structures in behavioral presentation and simulation.

Sierhuis, Clancey, Damer, Brodsky, and von Hoof in Chapter 8 provide a review of their development of an integrated virtual environment simulation system that incorporates lower level aspects such as finite state automata-based behavioral modeling with higher level reasoning and environmental modeling. The BrahmsVE system that they describe illustrates not only how interactive agents could be used for *in situ* training during long duration space flight, but also

how a specific programming architecture can be used to create such a system. In particular they incorporate a subsumption architecture as a flexible approach to behavioral modeling, but they connect it with lower level script-based modeling of kinematics and higher level modeling using belief-desire-intention architecture. Because of the attention to modeling at several different levels, their approach is similar to that described by Kruk in Chapter 3, though they stress the higher level constraints on the agents behavior with explicit descriptions of how such constraints may be abstracted from observation of actual behavior.

We are reminded by Kosinski and Zaczek-Chrzanowska's Chapter 9 on behaviorism of one of the original scientific approaches to an analysis of complex, intelligently driven behavior. In an attempt to mimic the success of the physical and biological sciences in the 19th and early 20th Centuries, Pavlov and later Behaviorists used a reductionist approach and tried to analyze all behavior into chains of unconditioned and conditioned responses. Though the empirical approach they brought to the study of simple and intelligent behavior represented a significant advance over previous 'arm-chair" philosophical approaches, they did not ultimately incorporate much of the syntactic, contextual factors and logical analysis that must be incorporated for a full description of intelligence. Nevertheless, as Kosinski points out, some of the behaviorist ideas have been incorporated into simple robotic learning systems and enable such systems to demonstrate some elements of intelligent or at least adaptive behavior.

Kurzweil in Chapter 10, *The Emergence and Impact of Intelligent Machines*, does not speak directly to what intelligence is or what makes intelligent machines seem intelligent, but rather makes an argument that whatever technological limits may currently make supposedly intelligent machines fail to live up to their billing, the inexorable, geometrically waxing flood of information technology will enable all defects to be resolved and thereby make the appearance of intelligent agents as embodied robots or as chimerical avatars inevitable. In a sense his argument about the inexorable progress of technology may make controversies about the true or correct way to represent human intelligence moot. The press of technological advance may make even suboptimal approaches successful. He argues that those who watch a technology develop regularly fail to recognize the interacting geometric growth of mutually supporting technologies. An example would be the interacting development of microcircuitry and personal computers. Observers consequently tend to see technological growth erroneously as a linear process. The chapter included is excerpted from a larger paper about the future of nanotechnology but provides examples from a variety of areas to support his main point.

Chapter 11 by Alexander and Ellis provides a discussion of some of the material presented at the meeting, which is included in the following ten chapters. They assess issues associated with giving users of virtual environments a sense of interaction with embodied, intelligent agents.

References

Brooks, R. (1986). A Robust Layered Control System for a Mobile Robot. *IEEE Journal of Robotics and Automation,* 2 (1).

Chapter 1
Constraint, Intelligence, and Control Hierarchy in Virtual Environments

***

Thomas B. Sheridan
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139
U.S.A.

1. Introduction

This paper seeks to deal directly with the question of what makes virtual actors and objects that are experienced in virtual environments seem real. (The term *virtual reality*, while more common in public usage, is an oxymoron; therefore *virtual environment* is the preferred term in this paper).

Reality is difficult topic, treated for centuries in those sub-fields of philosophy called ontology—"of or relating to being or existence" and epistemology—"the study of the method and grounds of knowledge, especially with reference to its limits and validity" (both from Webster's, 1965). Advances in recent decades in the technologies of computers, sensors and graphics software have permitted human users to feel *present* or experience *immersion* in computer-generated virtual environments. This has motivated a keen interest in probing this phenomenon of presence and immersion not only philosophically but also psychologically and physiologically in terms of the parameters of the senses and sensory stimulation that correlate with the experience (Ellis, 1991). The pages of the journal *Presence: Teleoperators and Virtual Environments* have seen much discussion of what makes virtual environments seem real (see, e.g., Slater, 1999; Slater et al. 1994; Sheridan, 1992, 2000).

Stephen Ellis, when organizing the meeting that motivated this paper, suggested to invited authors that "We may adopt as an organizing principle for the meeting that the genesis of apparently intelligent interaction arises from an upwelling of constraints determined by a hierarchy of lower levels of behavioral interaction." My first reaction was "huh?" and my second was "yeah, that seems to make sense." Accordingly the paper seeks to explain from the author's viewpoint, why Ellis's hypothesis makes sense. What is the connection of "presence " or "immersion" of an observer in a virtual environment, to "constraints" and what types of constraints. What of "intelligent interaction," and is it the intelligence of the observer or the intelligence of the environment (whatever the latter may mean) that is salient? And finally, what might be relevant about "upwelling" of constraints as determined by a hierarchy of levels of interaction?

2. Optimization as Related to Constraint

In theory, greatest goodness, happiness, degree or success, or, in technical parlance, utility, are a matter of simultaneous solution of two kinds of equations:

1) *Objective (utility) function:*
Goodness of solution = an explicit function of salient variables, e.g., dollars, time, accuracy, etc.)

2) Certain *given constraint equations* apply, for example:
- The dollars spent must be less than some total budget
- The time spent must occur before some deadline
- The performance = some explicit function of all the salient variables (e.g., dollars, time, constants.)

Formally, the optimal solution is found by maximizing goodness (utility) under the given constraints. This operation is standard practice in control engineering, in operations research, in econometrics, in engineering design, and in any endeavor where it is possible to express the objective function and the constraints in mathematical form.

While we cannot be so explicit about the mathematics and therefore declare optimality, efforts to design good environmental interactions for humans must take explicit account of constraints in much the same manner as above. In fact it will become apparent that what we call "intelligent" and "realistic" is precisely because certain environmental interactions that we have evolved and refined (i.e., optimized, at least in the sense of *satisficed* (made good enough for practical present use in our everyday life) conform to certain constraints. Examples will be cited in:

- Natural language
- Music
- Body movement (dance and athletics)
- Computer programming and supervisory control
- Computer displays

…and finally, by analogy,

- Virtual environments

2.1 Constraints of Spoken and Written Natural Language

Think about natural language. Random words spoken or written are just babble, and communicate nothing, even though with randomness their Shannon information measure ($\Sigma p_i \log p_i$, where p_i is the probability of speaking any particular word) is maximum. Spoken language makes "sense" to us precisely because it conforms to rules of syntax. Chomsky has shown, in fact, that these rules – constraints – are to a great extent programmed into our genes.
This is true of both spoken speech and writing. In fact we have evolved many additional constraints in order that we may communicate "meaning" to one another (grammar, spelling, presentation format). Without adherence to accepted constraints we simply do not understand the message, and even small errors detract and are regarded as foreign or unintelligent. Language is a coding scheme shared by the sender and the recipient, and without compliance to the constraints of the code there is no way for the recipient to select the meaning that the sender intends.

2.2 Constraints of Music

There is nothing much rational about music. “Good” music is what people like—esthetically. Random sound is not interesting. It is just noise. Music must adhere to constraints in frequency range (pitch) and sound intensity (loudness) in order to be heard at all and not damage our organs of hearing. These ranges are constrained to only a few octaves only a few dB respectively. The constraint of harmony makes it interesting (disharmonies are used only sparingly). Perhaps less obvious is the constraint of tempo (or rhythm or beat) of different notes: between 1 and 5 Hz. Slower or faster tempos would not even be considered music.

2.3 Constraints of Body Movement: Dance and Athletic Games

To be called dance body movements must be “graceful” (meaning smooth continuous motion of extended limbs). It must conform to a natural tempo of the body, one corresponding to natural frequencies of flaccid limbs (again between 1 and 5 Hz). Otherwise it is boringly slow (maybe Tai Chi goes to 0.2 Hz) or at too “forced” a pace.

Athletes, in order to make strong and precise movements (whether in running, jumping, throwing, hitting with a bat or racquet, swimming or gymnastics) must abide by constraints of the body (limb forces and reaction times), again in the 1-5 Hz range. Of course they are also bound by the constraining rules of the game (established to be consistent with the body’s own constraints on what is achievable.

2.4 Constraints in Design of Computer and Graphic Displays

Many studies have shown that when too much information is presented in a computer display much less information is communicated than if there were less information (i.e., fewer words or graphics on the same page or screen, less clutter). We all recognize the tendency to overdo the PowerPoint slide. It is apparently done because we can: The application program makes it so easy to add great variety in color, line widths, shading, logos, etc. This addition of what Tufte (1982, 1997) aptly calls “chart-junk” serves no good purpose, usually confuses the observer and actually reduces the effectiveness of communicating the essential message.

The chaos of trying to interpret the alarm tiles in a nuclear power plant control room during a emergency is often cited as an example of too much information too fast. In a test conducted at a nuclear plant simulator at one minute after a coolant major coolant pipe break I counted over 1000 alarm tiles lit up, and after a second minute 800 more. The nuclear plant operators claim that under such circumstances they have to rely on what they call “pattern recognition” to make any sense of what is going on. Clearly too much information, probably much of it inappropriate to the situation, is being presented for the constraints of human perception/recognition (mostly because too little thought has gone into integrating the information). (Traditionally each alarm tile is independently engineered to specify abnormality of every measured variable, and there is inadequate coordination among suppliers).

In a simple laboratory experiment in selecting one of a small number of control actions in view of uncertain information and time pressure, Roseborough (1988) showed that it was much better to display only point estimates than two-dimensional probability density functions.

Moral: Less can be more; constraints can be helpful.

2.5 Constraints of Computer Programming and Supervisory Control

Computer programming is accomplished in conformance to a highly constrained software language and operating system. The slightest discrepancy in programming disables the program with respect to its intended purpose, often showing up late in the process as a "bug."

Supervisory control (Sheridan, 1992) means a human communicates task goals and constraints to a computer that in turn works within already programmed task goals and constraints (as well as constraints of a dynamic electromechanical system) to perform the given task. This form of control is characteristic of telerobots, automated factories, automated vehicles such as modern commercial aircraft, and many other modern systems. Feedback is via a highly constrained (and usually abstract) translation of events. The paradigm of supervisory control is shown in Figure 1 as a four level hierarchy of control loops. Each block is the controller for that block immediately below. The downward arrows from any block are control commands for the block immediately below, and the upward arrows from the lower block are the feedback to the upper block. The properties (capabilities) of each lower block are the constraints on the block immediately above. The computer block is "intelligent" to the human operator insofar as it does a good job of controlling what is subordinate to it and provides feedback that conforms to the intentions and expectations of the human operator. We might say that the actuator agent appears "intelligent" to the computer insofar as it does a good job of controlling the task and provides feedback that conforms to its intentions and expectations.

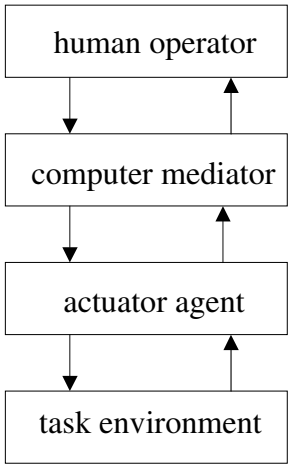


Figure 1. Hierarchy of supervisory control. (Each of computer, actuator and task environment imposes its own constraints. Constraints "well up" from below.)

Such a hierarchical paradigm applies not only to the human supervison of computer-mediated systems such as those mentioned above, but also to other forms of human interaction with the environment, where the blocks subordinate to the human are of other forms. For example in a natural language conversation between persons A and B (Figure 2), the second block down might represent the speaking/hearing constraints imposed on the cognition of the speaker A, which in turn feeds the hearing and gets feedback from the speaking constraints of B. From A's speaking perspective each lower block represents subsidiary constraints.

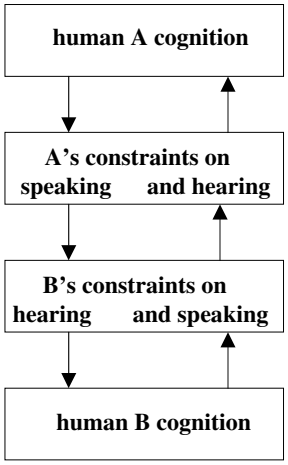


Figure 2. Constraints of interaction in social communication.

Having considered how human interaction with an environment, whether inanimate or social, succeeds as a function of conformance to the existing constraints, let us turn now to virtural environments.

3. Constraints That Apply in Virtual Environments

Seven types of constraints that apply in virtual environments are:

1. Sensory range and resolution of observer (absolute and differential thresholds)
2. Observation point consistency in space and time
3. Continuity of kinematics in space and time
4. Cause and effect
5. Mechanical impedance interaction with the observer/user
6. Symbolic interchange (words, gestures)
7. Etiquette (application of Grice's four maxims)

This is surely not the only credible taxonomy nor does it purport to be a complete set of constraints. There are many other factors that constrain what a virtual environment must be, given today's technology and today's human observers.

3.1. Constraints on Sensory Range and Resolution

This is the psychophysical set of constraints. A "sensorium" such as is depicted in Figure 3 is a region in the space of salient stimulus variables (in whichever of the five usual sensory modes or the much larger number of modes classified by Boring (1950) that can be sensed). While stimuli are commonly specified by intensity and frequency, there can be many other dimensions depending on the sensory mode, such as spatial distribution on the retina or skin, chemical makeup of the olfactory and gustatory stimulus, and temporal patterns for all the senses.

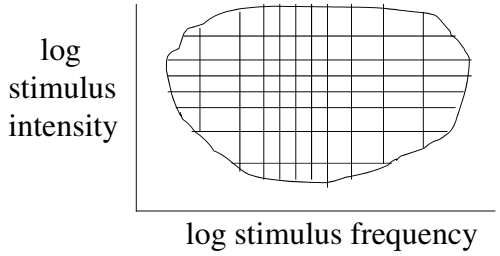


Figure 3. Stimulus sensorium in two of the (possibly many) dimensions.

3.2 Observation Point Consistency in Space and Time

The observer of a virtual environment often changes the viewing distance and angle of viewing. Indeed, to achieve a sense of presence and immersion relative to an object, such a change is often necessary. This is provided, of course, that the object size and orientation and lighting relative to the light sources remain geometrically correct during and following such a change. Achieving that correctness is a key task of the virtual environment designer.

If the observer is wearing a head-mounted display (HMD) it is important that the seen image correspond immediately to head position. Older HMD virtual environment systems exhibited a very noticeable time delay between when the observer suddenly turned his head and when the proper image appeared. Even current systems have small delays due to delays in the head-tracking hardware and image generation software.

3.3 Constraints on Continuity of Kinematics in Space and Time

Kinematics (relative motion of body or object segments) must be continuous in space and time and true to what is being simulated. Because the real world is continuous, virtual visual images that appear jerky will be discounted by the observer immediately. An infinitely high resolution in space (infinite number of pixels and polygons) and time (infinitely high speed) is obviously not possible and not necessary, but these resolutions must approach thresholds of discrimination in order to appear real. We are not there yet with today's computers, but getting close. Achieving a sufficient degree of continuity is probably the greatest challenge with respect to computer hardware.

3.4 Constraints on Cause and Effect

Dynamic interaction between the observer and sensed objects, as well as such interactions between two or more elements in the virtual environment, should play out with a cause-and-effect relations consistent with the known laws of physics. For example, if moving object A collides with fixed object B (either because the observer is controlling A or A is seen to be moving on its own), then B should either move or be dented and A should bounce off. If a liquid spills it should splash, and if an object is dropped into a liquid it should make proper waves, etc.

3.5 Constraints on Mechanical Impedance Interaction with the Observer

Newton's law says that for any action there is an equal and opposite reaction, and there are other well known laws of mechanics. Accelerating a mass, sliding an object along a friction surface, or compressing a spring—all result in a force imposed back on the limb or real object doing the moving. A realistic virtual environment would provide force feedback in a corresponding manner—both resolved kinesthetic force to the muscles and joints (as provided by a master manipulator arm) and tactile patterns (forces distributed in time and space, as provided by a hand worn tactile display such as the Dataglove). Current virtual environment systems may provide one or the other, but not both in combination.

3.6 Constraints on Symbolic Communication with an "Intelligent" Entity: Etiquette

An advanced virtual environment allows for meaningful symbolic dialog with a virtual "intelligent being"—either by speech or gesture or written language—much as in a Turing test (where the challenge is to have a computer-based conversation with a human such that the human cannot distinguish the computer from another human). This moves us beyond the typical image of a virtual environment as one of a static physical space and into one that includes other intelligent entities that can interact linguistically with the user/observer. These entities may appear to be humans of a known and present culture, they could be humans of a different culture from the past, or they could be imagined intelligent robots from outer space in a science fiction virtual environment.

Beyond the kinematic/dynamic constraints on gesture, or the grammar constraints on speech or written language, the intelligent being must comply with the constraints of etiquette —the social conventions that have evolved in civilized societies over centuries to enhance cooperation and smooth the communication interactions between people.

Grice (1975) postulated four maxims for cooperation in human-human conversation:

1. Maxim of quantity: Say what serves the present purpose but not more.
2. Maxim of quality: Say what you know to be true based on sufficient evidence.
3. Maxim of relation: Be relevant, to advance the current conversation.
4. Maxim of manner: Avoid obscurity of expression, wordiness, ambiguity, and disorder.

Miller (2000) has applied Grice's axioms to human-computer cooperation, especially adaptive user interfaces. His rules, in abbreviated form are:

1. Make many conversational moves for every error made.
2. Make it very easy override and correct any errors.
3. Know when you are wrong, mostly by letting the human tell you.
4. Don't make the same mistake twice.
5. Don't show off. Just because you can do something does not mean you should.
6. Talk explicitly about what you are doing and why. (Your human counterparts spend a lot of time in such meta-communication.)
7. Use multiple modalities and information channels redundantly.
8. Don't assume every user is the same; be sensitive and adapt to individual, cultural, social, and contextual differences.
9. Be aware what your user knows, especially what you just conveyed (i.e., don't repeat yourself).
10. Be cute only to the extent that it furthers your conversational goals.

Grice surely had in mind sentient human beings of the same culture, while Miller seems to have had in mind a computer decision aid, where "presence" or "reality" have more to do with the intelligence of the computer than with its appearance. My purpose here is to make a distinction between appearance of reality (immersion) and intelligence. A virtual brick can appear real if we can view it from any desired angle and distance, handle it, etc, but the brick is hardly intelligent. A virtual computer that looks like a computer and responds like a computer will not seem to be other that a computer. But if the computer looks like a computer but responds like a human being, a user will be drawn in to a conversation as if there were a human on the other side. In other words there can be "presence" and "immersion" based on the apparent intelligence of the other entity, rather than physical appearance

4. Computer Assistance in Designing Virtual Environments

Designing a virtual environment is a problem in hierarchical control and in design. In the abstract, control and design are the same: one has goals that need to be specified, and constraints that need to be fulfilled. Specifying the goal really amounts to specifying the objective function—the tradeoff between salient variables. Specifying the constraints amounts to thinking hard about the limitations on the physics involved as well as the physiological and psychological properties of the users. Unfortunately, rarely can these equations be specified mathematically—as is required to derive a unique optimal solution. This still does not relieve the designer of the need to struggle with explication of the tradeoffs between objectives and the constraints—typically many and diverse, as has been suggested in the foregoing paragraphs.

Often the constraints on a virtual environment can be stated as single variable limits: a minimum number of polygons, a minimum refresh rate, a restricted range of apparent distances and viewing angles and rate of change of viewpoint within this space, a few key physical interaction phenomena between elements of the virtual scene, a restricted stimulus (e.g., vision, tactile

sensing but no sound) which is expected to provide sufficient sense of reality, etc. Sometimes there will also be known relations between two or a few variables, e.g., a tradeoff between number of polygons and refresh rate These constraints can then be put into a computer, and the constraining relations will bound a space of hopefully not too many variables. However, as with any real design problem, the number of variables will be variables but usually significantly greater than three – so that visualization of the bounded (feasible) solution space is not possible by an ordinary mortal (whereas the computer has no problem).

Such a hyperspace will be characterized by many bounding hyperplanes, some perpendicular to the relevant variable, others at angles when there is a linear tradeoff relation between variables (or a surface where the relation is nonlinear). There will be many "corners" of this feasible solution space where planes and surfaces intersect. If all constraints are constants (planes perpendicular to axes) or linear relations (planes at angles) this is essentially the solution space of linear programming. Figure 4 shows a very simple example (in two variables) of how one or another of the corners tends to intersect the best objective function (tradeoff) curve of indifference between number of polygons and refresh rate. The computer can easily find this optimal solution in a much more complex hyperdimensional space—assuming there are a large number of alternatives available to select from in the feasible solution space and the tradeoff indifference lines are known.

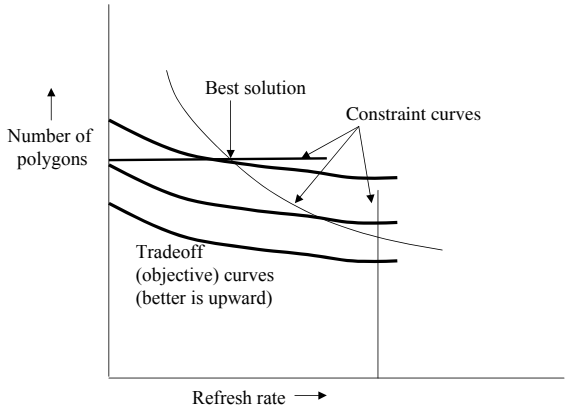


Figure 4. Best solution at intersection of constraint lines

Let us assume, however, there is only a small discrete number of solutions alternatives available (others perhaps being rejected by constraints) and one must select among these (again given that the tradeoff indifference lines are available). The computer can again be of assistance. Figure 5 shows this situation, in a very simple form. The four shapes represent four alternative designs. The triangle can be rejected immediately because it is dominated by the square (the latter being better in number of polygons and the equal in refresh rate. The cross, square and circle form a so-called Pareto optimal (non-dominated) set. However the square is the best because it is at the highest utility (the highest tradeoff indifference curve).

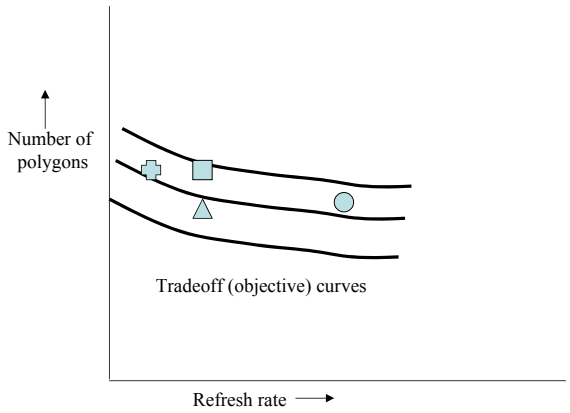


Figure 5. Best solution as the Pareto optimal alternative with the greatest utility.

Finally, if the tradeoff indifference curves are not known, the computer can at least throw out the alternatives that are dominated and leave the designer to choose among the alternatives an the Pareto frontier. In a complex design problem where the solution space cannot be visualized this alone can be of great help. Charny and Sheridan (1989) demonstrated these techniques for interacting with the designer in a five–dimensional solution space.

5. Conclusion

If the many expected constraints are not adhered to, a virtual environment (as with many other forms of human interaction) does not appear real or even intelligent. Explication of constraints as well as objective function (in the form of tradeoff indifference curves for salient variables) allows the computer to assist the designer in selecting a best design.

References

Boring, E.G. (1950). A history of experimental psychology. NY: Appleton-Century-Crofts.

Charny, L. and Sheridan, T.B. (1989) Adaptive goal setting in tasks with multiple criteria. In Proc. IEEE 1989 International Conference on Cybernetics and Society.

Ellis, S.R., Kaiser, K.K., and Grunwald, A.J. (eds.) (1991). *Pictorial communication in virtual and real environments 2nd Ed*., London: Taylor and Francis.

Grice, H.P. (1975). Logic and conversation, In P. Cole and J. Morgan, *Syntax and Semantics: Speech Acts,* vol. 3. NY: Academic Press.

Miller, C.A.(2000). Rules of etiquette, or how a mannerly AUI should comport itself to gain social acceptance and be perceived as gracious and well behaved in polite society. Unpublished working notes of AAAI Spring Symposium on Adaptive User Interfaces (March 20-22, 2000).

Roseborough, J. (1988). Aiding human operators with state estimates. PhD Thesis, Cambridge, MA, MIT.

Sheridan, T.B. (1992). Musings on telepresence and virtual presence. *Presence,* vol. 1, pp. 120-125.

Sheridan, T.B. (1992). Telerobotics, automation and human supervisory control. Cambridge, MA: MIT Press.

Sheridan, T.B. (2000). Descartes, Heidegger, Gibson and God: Toward an eclectic ontology of presence. *Presence,* vol. 8, no. 5, pp. 549-557.

Slater, M. (1999). Measuring presence: A response to the Witmer and Singer presence questionnaire. *Presence,* vol. 8, pp. 460-565.

Slater, M., Usoh, M., and Steed, A. (1994). Depth of presence in virtual environments. *Presence,* vol. 3, pp. 130-144.

Tufte, E.R (1997). Visual explanations. Cheshire, CT: Graphics Press.

Tufte, E.R. (1983). The visual display of quantitative information. Cheshire, CT: Graphics Press

Webster's Third new International Dictionary (1965). Springfield, MA: G. & C. Merriam Co.

Chapter 2
Hierarchically Structured Non-Intrusive Sign Language Recognition

***

Jorg Zieren and Karl-Friedrich Kraiss
Chair of Technical Computer Science
RWTH Aachen University
fzieren,kraissg@techinfo.rwth-aachen.de
www.techinfo.rwth-aachen.de

Abstract

This work presents a hierarchically structured approach at the nonintrusive recognition of sign language from a monocular frontal view. Robustness is achieved through sophisticated localization and tracking methods, including a combined EM/CAMSHIFT overlap resolution procedure and the parallel pursuit of multiple hypotheses about hands position and movement. This allows handling of ambiguities and automatically corrects tracking errors. A biomechanical skeleton model and dynamic motion prediction using Kalman filters represents high level knowledge. Classification is performed by Hidden Markov Models. 152 signs from German sign language were recognized with an accuracy of 97.6%.

1. Introduction

Manual gestures are an important information carrier in everyday communication. Considerable potential lies in the automatic recognition of gestures, especially for human-computer interaction. As opposed to the keyboard, gestures are natural, intuitive, and do not require special skills. Sign language recognition is a particularly challenging field in this research area. Its goal is to do for deaf people what speech recognition has done for hearing people: Offer the most natural way of controlling electronic devices.

In sign language, information is communicated primarily through hands and face. This work utilizes manual parameters for the recognition of 152 signs from German sign language. Instead of using data gloves, which in most scenarios is not an acceptable solution, manual parameters are extracted from images acquired by a single video camera positioned in front of the signer. Skin color and motion form the basic low-level image cues. In order to ensure a natural, i. e. non-intrusive interaction, no other devices, such as markers or additional cameras, are employed.

Existing non-intrusive systems only support considerably smaller vocabularies of about 40 signs [12]. Due to the difficulty of accurately localizing the signer's hands when they are overlapping with each other and/or with the face, which occurs frequently in sign language, ambiguities can easily arise. Sophisticated overlap resolution procedure and the parallel pursuit of multiple hypotheses regarding hands position and movement, are applied to compute manual features even in such problematic scenes. *A. priori* knowledge is incorporated through a biomechanical skeleton model and dynamic Kalman filter predictions.

The extracted features are classified using Hidden Markov Models (HMMs) to compensate for variations in speed and allow limited variations in amplitude. On the chosen data set the developed system achieves a recognition rate of 97.6% at a resolution of 384 *X* 288 pixels. This performance has been measured for a person dependent recognition task in a controlled environment. However, the system's basic concepts are not geared towards this scenario. Their suitability for "real life", possibly mobile environments, is an important design feature.

2. Sign Vocabulary

The system's vocabulary consists of 152 signs. Each has been recorded ten times with a resolution of 384 *X* 288 pixelS and 25 frames per second. Figure 1 shows an example sign and the recording conditions, which were identical for all signs. Since this work focuses on person-dependent classification, only one signer has been recorded. Extending the system to person-independent classification would not affect the tracking stage, but inter-personal variance would require special measures in the classification stage if comparable recognition rates were to be achieved.

Figure 1. Example sign "computer" from the test/training data set. Signer, background, illumination, and clothing were identical in all recordings.

3. Tracking

The complexity of the object tracking task suggests a hierarchical division in two stages (see Figure 2). First, a low level processing stage detects a set of target candidates using skin color as an image cue. This set may include skin colored distracters. Hands and face are then found in this set by the subsequent high level processing stage. To this end, multiple hypotheses are evaluated per frame and over time.

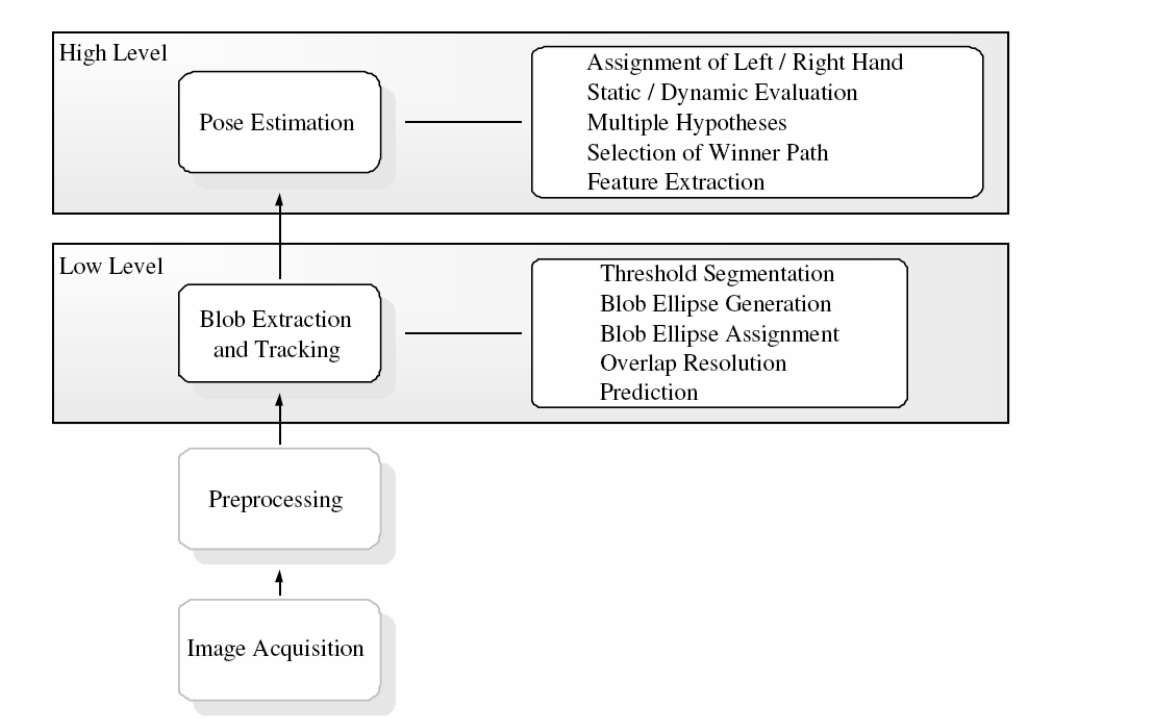


Figure 2. Functional overview.

3.1 Target Candidate Detection

Based on generic skin and non-skin color histograms presented in [7], a skin color probability is computed for every pixel. After smoothing the result with a Gaussian kernel and application of a threshold segmentation [13], contiguous regions (blobs) are extracted.

The computationally demanding high level stage necessitates efficient data structures for the representation of each blob. Therefore, a blob's boundary (which typically consists of several hundred pixels) is not processed directly, but approximated by an elliptical representation called "blob ellipse." Aiming for a tradeoff between accuracy in terms of the signal to noise ration and processing speed, a blob ellipse is described by its center coordinates x and y, radii r_a and r_b, and orientation of the principal axis.

It is obvious that a threshold segmentation cannot separate two or more overlapping skin colored objects (e. g. hand and face). To extract meaningful features, however, a separation of the overlapping objects is required. Therefore, a distinction is introduced between the set of "raw" blob ellipses extracted in frame t, called $B_{raw;t}$, and a corresponding set of "overlap resolved" blob ellipses B_t. Only B_t will later be forwarded to the high level stage. This is illustrated in

Figure 3. In the input image I_{t-1}, no overlap is present. Therefore, $B_{raw;t-1} = B_{t-1}$. In I_t, the right hand is overlapping with the face. In $B_{raw;t}$, the two corresponding ellipses have therefore merged into one. The low level stage resolves this overlap and computes two overlapping ellipses. This process is described in the following section.

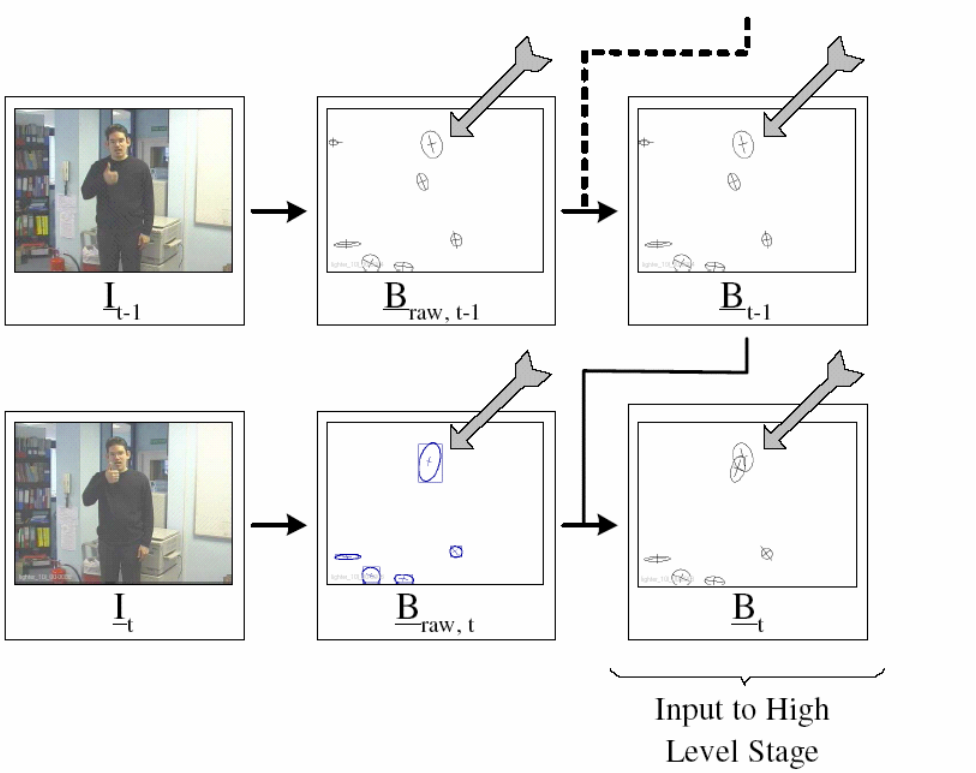


Figure 3. Processing of blob ellipses by the low level stage. Grey arrows indicate overlap resolution.

3.1.1 Overlap Detection and Resolution

For each blob ellipse in $B_{raw;t}$, a number of n an element of N_0 corresponding blob ellipses in B_{t-1} are found by computing and evaluating predictions for both shape and position. Depending on n, several cases can be distinguished as shown in Table 1.

Table 1.
Different cases of blob ellipse correspondence.

n = 0	new object has entered the image
n = 1	regular tracking
n ≥ 2	*n* objects have started to overlap

For n ≥ 2, either the EM or the CAMSHIFT algorithm is used to resolve the overlap and approximate features for all overlapping objects. This is described below.

3.1.2 Overlap Resolution Using the EM Algorithm

The Expectation Maximization (EM) algorithm is an iterative method for approximating a given probability distribution by a superposition of a fixed number of two-dimensional Gaussian distributions [2]. The latter corresponds well with the concept of blob ellipses, which allows an easy integration of the EM algorithm in the processing chain.

Figure 4 shows a typical scenario that can be treated with the EM algorithm. The overlap here is only partial, i. e. none of the three overlapping objects is completely enclosed in another object (in a 2D sense). Since the threshold segmentation yields a binary mask, but the EM algorithm computes a superposition of Gaussians, a morphological distance transformation (described in [6]) is used to create a pseudo-multivariate distribution. This requires that non-skin colored pixels (holes) enclosed by the overlapping objects are first removed, i. e. set to 1 in the binary mask.

Figure 4. Preparation of the skin color mask for the EM algorithm. (a) Original image, (b) Threshold segmented skin color probability, (c) Skin color mask with holes removed, (d) Distance transformed skin color mask.

For the EM algorithm to accurately resolve an overlap of multiple blob ellipses, t is initialized with the shape and position parameters computed for these ellipses in the previous frame. The original algorithm has been modified so that several parameters either remain constant or change only in a well-defined interval. This increases the stability of the approximation process. Figure 5 shows the approximation status at different iterations.

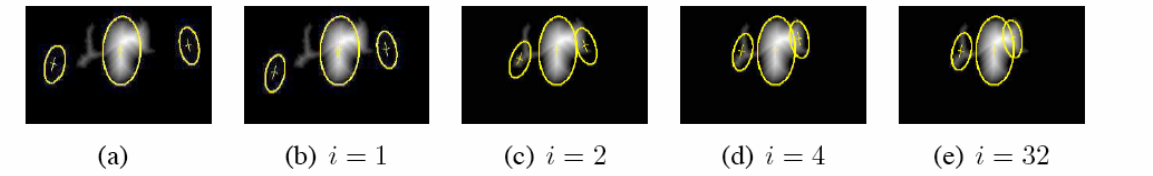


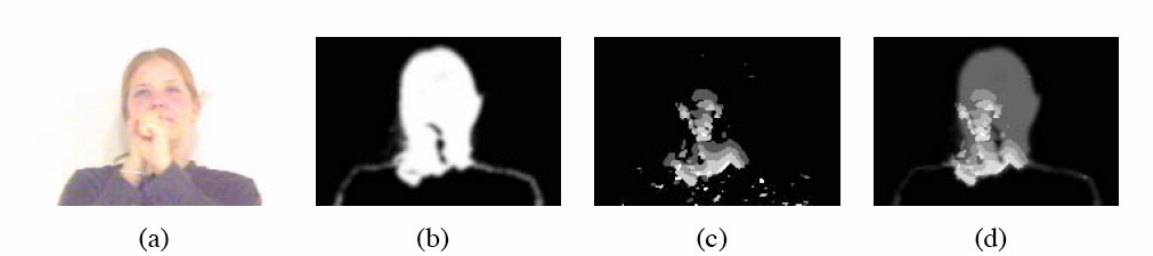
Figure 5. Application of the EM algorithm. (a) Initialization, (b – e) after *i* iterations.

3.1.3 Overlap Resolution Using the CAMSHIFT Algorithm

If one object is completely enclosed in another, the EM algorithm is unsuitable for overlap resolution because its input matrix (Figure 4d) would not provide any information about the inner object. However, motion can be used as an additional image cue in this case. The detection of motion is based on computing, for every pixel, the difference in color between successive frames, and subsequent application of a fixed threshold to yield a binary "motion mask." Using a sliding average with linearly decreasing weights, a so called Motion History Image (MHI), $I_{motion}(x; y)$, is created as described in [3]. This is then combined with the skin color probability distribution $I_{skin}(x; y)$ according to the following equations. Figure 6 shows a visualization of this process.

$$I_{skin}(x; y) = p_{skin}(x; y) \quad (1)$$

$$I_{camshift}(x; y) = w\, I_{skin}(x; y) + (1 ¡ w)\, I_{motion}(x; y) \quad (2)$$



(a) (b) (c) (d)

Figure 6. Computation of the CAMSHIFT input image. (a) Original image, (b) Skin color probability image (Gauss filtered), (c) Motion History Image, (d) Combined image according to equation 2.

On the resulting image $I_{camshift}$, a CAMSHIFT tracker is applied for each hand. The respective search windows are initialized with the most recent position and shape values of the overlapping blob ellipses.

Shape and orientation remain constant while this method is used for overlap resolution. The weight w ($0 \cdot w \cdot 1$) allows adjustment of the degree to which motion is considered by the CAMSHIFT algorithm. Figure 7 shows an example application to a face-hand-hand overlap.

(a) t=13 (b) t=18 (c) t=23

Figure 7. Resolution of a face-hand-hand overlap by application of the CAMSHIFT algorithm.

3.2 Multiple Hypothesis Tracking

From the set of detected target candidates, the actual body configuration that gave rise to this observation has to be deduced. Since every observation allows more than one interpretation, multiple hypotheses can be formulated for every video frame. Enumerating these hypotheses and plotting them over time results in a diagram as shown in Figure 8. In this hypothesis space, there are $N(t)$ hypotheses for frame t, thus the total number of all possible paths (i.e. tracking results) P equals

$$P = \prod_{t} N(t) \qquad (3)$$

High level knowledge about the signing process allow computing for each hypothesis a probability $p_{stat;t}(i)$ which is independent from the previous and the next hypothesis (static), and a probability $p_{dyn;t}(i; j)$ which depends only on the transition between two hypotheses, but not on the hypotheses themselves (dynamic). Searching for the path with highest total probability is done with the Viterbi algorithm [9].

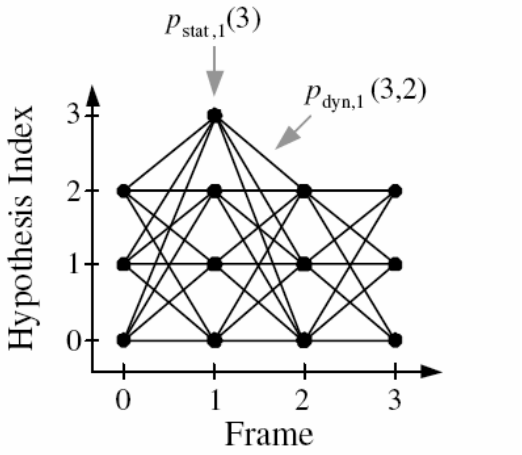


Figure 8. Hypothesis space with static and dynamic probabilities.

3.2.1 Computation of Static and Dynamic Probabilities

The chosen approach allows exploitation of any number of image cues and high level knowledge for the computation of the static and dynamic probabilities. In practice, the selection will depend on the actual recognition/tracking task. For the presented system, a body model is computed that approximates arm length and flexion of joints. From a manual segmentation of the input clips, a hand position histogram has been created that represents knowledge of where the hands are typically found. Together with information about the signer's handedness, this allows to evaluate the hypothesized configuration's likeliness both physiologically and "linguistically," resulting in p_{stat}. p_{dyn} is obtained from Kalman filter predictions for each blob ellipse's position, shape, and orientation.

3.3 Feature Vector Composition

The feature vector composed for every frame contains the center coordinates, orientation, area, and ratio of radii of each hand's elliptical approximation. Furthermore, compactness and eccentricity (as defined in [10]) are computed from the object's border found by the threshold segmentation. For the feature vector to be independent from the signer's exact position in the image and from the camera's resolution, position and area are specified relative to the face position and face width. Derivatives for all of these values are also added to the feature vector.

4. Evaluation

Since the tracking stage is the most complex component in this work, not only the recognition rate, but also the tracker's hit rate were evaluated. A manual segmentation of all input clips has been performed which allows to define three categories that classify a tracking result based on the center coordinates as shown in Table 2 and Figure 9.

Table 2. Categorization of tracking results.

Center coordinates (*x, y*) within...	Category
...border of the target object	hit on object
...elliptical region around target center	hit near center
...neither of the above	miss

Figure 9. Definition of tracker hit and miss.

Experimental results are shown in Table 3. The vocabulary has been divided into five categories that clearly show overlap to be the tracker's main problem.

Table 3. Quantitative evaluation of tracking accuracy (H=hand, F=face).

Sign Category	One Handed	Two Handed	No Overlap	H-F Overlap	H-H Overlap	Total
Hit rate:	98.4%	95.4%	99.0%	96.8%	93.7%	97.1%

On the complete vocabulary of 152 signs in German Sign Language, a recognition rate of 97.6% was achieved using an HMM based classification stage. This constitutes an increase compared to other recognition systems (intrusive and nonintrusive), such as [4], [5], [11], and [12]. Only for considerably smaller vocabularies (approx. 40 signs) have higher rates for non-intrusive recognition been published. This may be due to the fact that the multiple hypotheses approach considers for every frame nearly all available information, including past and future frames, before a decision is made, and can retrospectively correct tracking errors as soon as they become apparent.

5. Outlook

Several improvements and extensions are conceivable to either increase recognition performance or open up new application scenarios. A user adaptive skin color model would reduce the number of distracters by narrowing down the target color range and thereby increasing both reliability and processing speed of the tracking stage.

Significantly increasing the vocabulary size would require the extraction of further shape and/or texture features, with the ultimate goal of reconstructing a 3D hand model from the 2D image data.

Recognition of continuous signing is an obvious but complex extension of the system. Translation systems to speech, text, or another sign language, require the automatic detection of start and end points of individual signs, as well as the handling of co-articulation effects that can have strong influence on the extracted features.

Integration of mimic, i. e. facial features, is currently in progress. Facial expressions are vital for sign language recognition since many signs are identical in their manual features. A further increase in recognition rates can be expected from this extension.

Acknowledgements

This work was carried out at the Chair of Technical Computer Science, RWTH Aachen University, based on a dissertation by Suat Akyol [1]. Numerous other researchers and students have contributed code to the developed software [8]. The project is funded by the European Commission Directorate – General Information Society Technologies (IST) Programme (2001–2003)

References

[1] S. Akyol. *Nicht-intrusive Erkennung isolierter Gesten und Gebaerden (Non-Intrusive Recognition of Isolated Gestures and Signs)*. 2003. Dissertation, Chair of Technical Computer Science, RWTH Aachen University.

[2] J. A. Bilmes. A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussion Mixture and Hidden Markov Models. Technical Report TR-97-021, International Computer Science Institute, U.C. Berkeley, April 1998.

[3] J.-W. Davis and A.-F. Bobick. The Representation and Recognition of Action Using Temporal Templates. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pages 928–934, San Juan, Puerto Rico, 1997.

[4] K. Grobel. *Videobasierte Geb¨ardenspracherkennung mit Hidden-Markov- Modellen (Video-Based Sign Language Recognition Using Hidden Markov Models)*. Fortschritts-Berichte VDI 10/592. VDI Verlag, D¨usseldorf, 1999. Dissertation, Chair of Technical Computer Science, RWTH Aachen University.

[5] H. Hienz. *Erkennung kontinuierlicher Geb¨ardensprache mit Ganzwortmodellen (Recognition of Continuous Sign Language Using Whole Word Models)*. Shaker Verlag, Aachen, 2000. Dissertation, Chair of Technical Computer Science, RWTH Aachen University.

[6] A. K. Jain. *Fundamentals of Digital Image Processing*. Prentice-Hall Inc., Englewood Cliffs, NJ, 1989.

[7] M. J. Jones and J. M. Rehg. Statistical Color Models with Application to Skin Detection. Technical Report CRL 98/11, Compaq Cambridge Research Lab, December 1998.

[8] LTI-Lib: A C++ library for image processing and computer vision. http://ltilib.sf.net, 2003.

[9] L. Rabiner and B.-H. Juang. An Introduction to Hidden Markov Models. *IEEE ASSP Magazine*, 3(1):4–16, 1986.

[10] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis, and Machine Vision*. Brooks/Cole Publishing Company, 1999.

[11] T. Starner, J. Weaver, and A. Pentland. Real-Time American Sign Language Recognition Using Desk andWearable Computer Based Video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(12):1371–1375, 1998.

[12] M.-H. Yang, N. Ahuja, and M. Tabb. Extraction of 2D Motion Trajectories and its Application to Hand Gesture Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(8):1061–1074, 2002.

[13] J. Zieren, N. Unger, and S. Akyol. Hands Tracking from Frontal View for Vision-Based Gesture Recognition. In L. van Gool, J. Hartmanis, and J. van Leeuwen, editors, *Lecture Notes in Computer Science LNCS 2449*, Z¨urich, Switzerland, 2002.

Chapter 3
Intelligent Entity Behavior Within Synthetic Environments

***

R.V. Kruk, P.B. Howells, and D.N. Siksik
CAE Inc.
St. Laurent, Quebec, Canada

Abstract

This paper describes some elements in the development of realistic performance and behavior in the synthetic entities (players) which support Modeling and Simulation (M&S) applications, particularly military training. Modern human-in-the-loop (virtual) training systems incorporate sophisticated synthetic environments, which provide:

1. *The operational environment, including, for example, terrain databases;*
2. *Physical entity parameters which define performance in engineered systems, such as aircraft aerodynamics;*
3. *Platform/system characteristics such as acoustic, IR and radar signatures;*
4. *Behavioral entity parameters which define interactive performance, including knowledge/reasoning about terrain, tactics; and,*
5. *Doctrine, which combines knowledge and tactics into behavior rule sets.*

The resolution and fidelity of these model/database elements can vary substantially, but as synthetic environments are designed to be compose able, attributes may easily be added (e.g., adding a new radar to an aircraft) or enhanced (e.g. Amending or replacing missile seeker head/ Electronic Counter Measures (ECM) models to improve the realism of their interaction).

To a human in the loop with synthetic entities, their observed veridicality is assessed via engagement responses (e.g. effect of countermeasures upon a closing missile), as seen on systems displays, and visual (image) behavior. The realism of visual models in a simulation (level of detail as well as motion fidelity) remains a challenge in realistic articulation of elements such as vehicle antennae and turrets, or, with human figures; posture, joint articulation, response to uneven ground. Currently the adequacy of visual representation is more dependant upon the quality and resolution of the physical models driving those entities than graphics processing power per Se.

Synthetic entities in M&S applications traditionally have represented engineered systems (e.g. aircraft) with human-in-the-loop performance characteristics (e.g. visual acuity) included in the system behavioral specification. As well, performance—affecting human parameters such as experience level, fatigue and stress are coming into wider use (via AI approaches) to incorporate more uncertainty as to response type as well as performance (e.g. Where an opposing entity might go and what it might do, as well as how well it might perform).

1. Introduction

Synthetic environments play a major role in modern military training systems. Over the years they have progressed from providing basic threat layout to incorporation of the cooperative elements necessary for collaborative training in land, sea and air applications. To a large extent, this has been made possible through advances in computer-generated simulation of military forces using modeling technology, networking capabilities and improved computer hardware [6].

Modern synthetic environments provide solutions for a wide range of training needs. Using knowledge-based systems and detailed mathematical modeling of physical systems, it is possible to build simulations where the manned simulator works in concert with the computer-generated entities. A simulator may be linked to various synthetic environments via networking protocols such as the High Level Architecture (HLA) [3], allowing the simulator to see and be seen by the computer-generated entities. Direct communications between simulator and entity may be achieved via the simulation of digital radio and messaging, thus enabling a trainee to request data updates from and submit direct requests to computer-generated allies.

In the aviation arena, there is a great deal of emphasis on air-to-air combat mission training. This is because aircraft very rarely go into combat alone. The computer-generated entities play an important role in providing a manned simulator with wingmen and/or coordinated opponents.

In the naval arena, synthetic environments provide for training in anti-submarine warfare (ASW), anti-surface warfare (ASuW) and search and rescue missions. Naval tactics such as screening and tracking coupled with classification strategies for naval contacts (naval identification criteria) as well as Link 11 communications capability combine to support multiple mission exercises, a role fulfilled by the computer generated entities in modern day training devices.

Within the ground forces environment, simulation training revolves around the battle, command and control, geography, environment and logistics for relatively large numbers of individual entities, which may be aggregated in a number of ways. Synthetic environments provide the terrain, tactics, allied forces, opponents and weather elements necessary for realistic friendly and hostile interactions.

2. A Note on Standards

An extensive range of standards has grown up around the definition of platform and system characteristics in Networked Military M&S. These include specification of how platform and system entities are to be accessed, which began in the development of Protocols for Distributed Interactive Simulation (DIS) and have been carried over into the High Level Architecture protocols [4, 8]. Within HLA, further entity specification is defined in the Object Models for Simulations (SOMs) and Federations (FOMs). Performance levels of models for aircraft, radars, ordinance, etc., can be clearly defined, are predictable, and the implementation in simulation at run time is deterministic. This is not the case for representation of human entities (such as individual soldiers).

There are currently no widely accepted standards for representation of high fidelity human entities in simulation. Human models should be generated and run through approaches fundamentally different from those used for simulation of engineered constructs such as vehicles - if for no other reason, because of the necessity to represent far more degrees of freedom in motion, as well as model individual and aggregate behavior which may include emotional effects and "Situation Awareness". Traditionally human behavior in large command and control training systems has been generated by role—playing human operators—in some cases 50 role players might support the training of a 10 person command team. Automating these characteristics in Synthetic Environments used for training remains a significant challenge to system designers.

3. Some Examples of Human Modeling/Simulation Problems

1. Slippage/penetration: A synthetic human's feet and hands appear to slip on surfaces as if on ice or go through surfaces because knowledge about the precise location of the surface is not inherent in the Human joint model—rates of motion and degrees of freedom do not match, and inherent flexibility of human torso/limbs is difficult to simulate.

2. Inability to climb objects and/or deal with minor terrain variation: Same problem set as 1, plus difficulty in defining borders of 3D objects in conventional simulation architectures.

3. Jerkiness: A synthetic human jerks in transitions between postures because of lags between limb activation, difficulties in representing joint degrees of freedom and flexibility (especially the spine) and balance is anomalous (synthetic entities often look like they are about to fall over).

4. Inability to grasp objects: Same problem sets as 1 to 3—hands are often modeled like clubs.

We are able to model and simulate humans within machines much more effectively, and the bulk of the content of the present paper concentrates on the state of the art for humans embedded within systems.

4. General Architecture

The general architecture of a simulation application is an assembly of frameworks, libraries and data organized in layers as shown in
Figure *1*. The framework that is the foundation of the architecture is the Virtual Environment Framework. It defines how all the pieces fit together and interact with each other [2,7]. The next layer is composed of the application frameworks. Each one of these frameworks isolates a particular domain to allow various implementations to be used. The relationship between these frameworks should be limited to interfaces, which define the functionality, but not the implementation. The third layer is a set of libraries provided by various manufacturers. They provide an implementation of their associated framework. The last layer is composed of initial values for the objects defined in their associated library. All these data are linked together within the scenario to be executed in real-time.

Figure *1* also presents a breakdown of application frameworks that are related to the synthetic tactical environment domain. The main framework of this domain being the computer generated forces. The other frameworks either extend the capabilities (an expert system framework to give realistic behavior to players) or provide complementary capabilities to increase the realism of the simulation (terrain and weather frameworks).

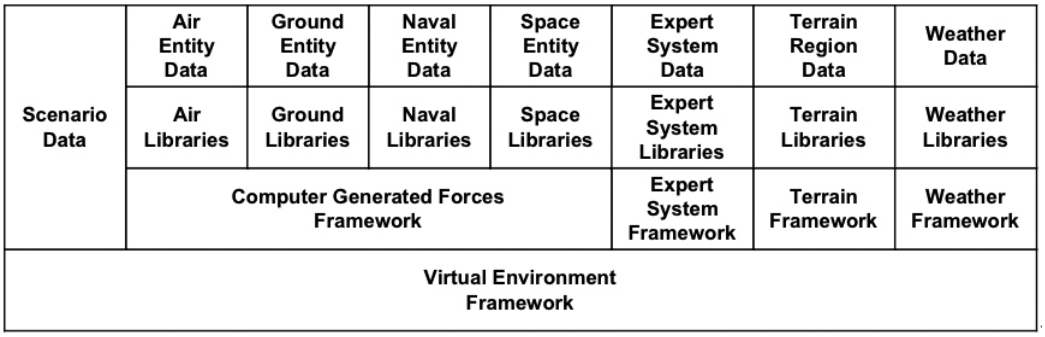


Figure 1. General Architecture of a Simulation Application

5. Virtual Environment Frameworks

The virtual environment framework implements the infrastructure of the general architecture. It is internally divided into three layers that provide various functions, as shown in Figure *2*.

The first layer, i.e. the foundation layer, includes the services necessary to support a truly expandable and configurable system. At the implementation level, it defines the plug-in concept to support libraries and other frameworks, and the data type concept to support attributes and object factories [1]. At the functional level, it defines the provider concept for adding global functionalities and the adapter concept for adding functionalities to individual objects. It also defines the interface with the repository and provides the default implementation.

The second layer, i.e. the execution layer, handles the publish, subscribe and ownership mechanisms. It defines interfaces with the scheduler and with the network. It also provides the default implementations for both. Within the context of this layer, each object performs its own publish and subscribe operations. The framework combines these operations before making similar operations through the network interface. An additional service allows access to the internal object information for verification and validation purposes.

The top layer, i.e. the distribution layer, defines how scenarios are distributed across multiple processes, and which may be executed on one or more computers. Each scenario includes initial parameters for their configurable elements of the framework such as the scheduler. When executing, the scenarios become exercises.

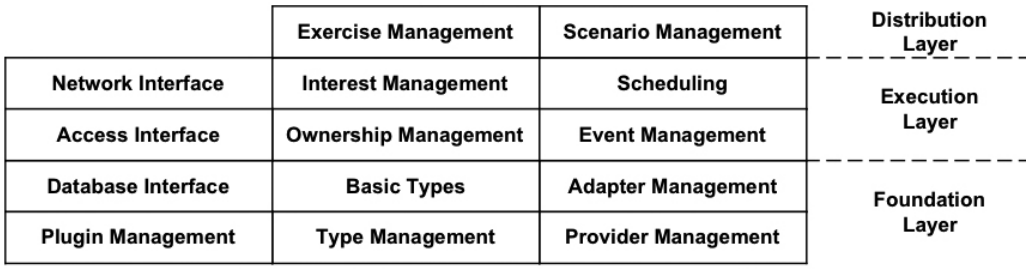


Figure 2. Virtual Environment Framework

6. Synthetic Environment (SE) Application Example 1

Air-to Air Combat with a legacy SE Tool: the Interactive Tactical Environment Management System (ITEMS) [5]

Maintaining a high degree of readiness in air-to-air combat operations is difficult. One alternative is to supplement airborne air combat training with simulator time. Effective training, however, requires that wingmen and opponents have representative characteristics and performance in terms of maneuvering, aircraft flight dynamics and decision processes.

Training can be supported in the areas of weapons deployment, basic fighter maneuvering, two-vs.-one and one-two-two engagements. For weapons training, the air target can be programmed to perform flight paths composed of simple weaves and turns, where the student is required to track the target and obtain a firing solution using guns or missiles. Control over the "g" capability of the target enables the instructor to intervene to make the task more or less difficult for the student. The simple maneuvers may be used for training in basic fighter maneuvering. The air target, for example, may be set to perform a pursuit maneuver which the manned simulator has to counter.

To support training in air-to-air combat at a more sophisticated level, ITEMS incorporates Interactive Air Targets (IATs). The air target is controlled by a programmable rule base. The rules are defined using condition and response parameters selected from a pull-down menu. For example, the condition parameters for the IAT to perform a guns engagement are for the opponent to be in the front hemisphere and in guns range. The response parameter in this case would be to fire the gun. The air target responds continuously to the rules that are triggered as the air target switches from offensive to defensive positions depending on the engagement geometry.

The air target considers the full equations of motion in the longitudinal axis with some simplifications in the lateral axis. The targets are defined using an off-line Database Management System (DBMS) for the entry of the principal aerodynamic data (lift and drag) and characteristics of the propulsion system. A range of weapon systems and sensors can be assigned to the aircraft. Typical aircraft types supported include F4-Phantom, Tornado, MiG-29 Fulcrum, and F15-Eagle, etc.

7. Components

a) Overview

The method of defining a tactical scenario is a bottom-up approach, where the elementary element databases (weapons, sensors, and countermeasures) of the scenario are first defined. These elements are then combined to generate higher-level databases that allow for a complete tactical environment by combining all elements and pertinent data. Access to the library of databases is provided via dedicated graphical user interfaces.

The principal components are illustrated in Figure 3. They include the air target definition, air combat maneuvering database, rules database, the weapons and sensor database, and the environment database. The function of each of these components is described below.

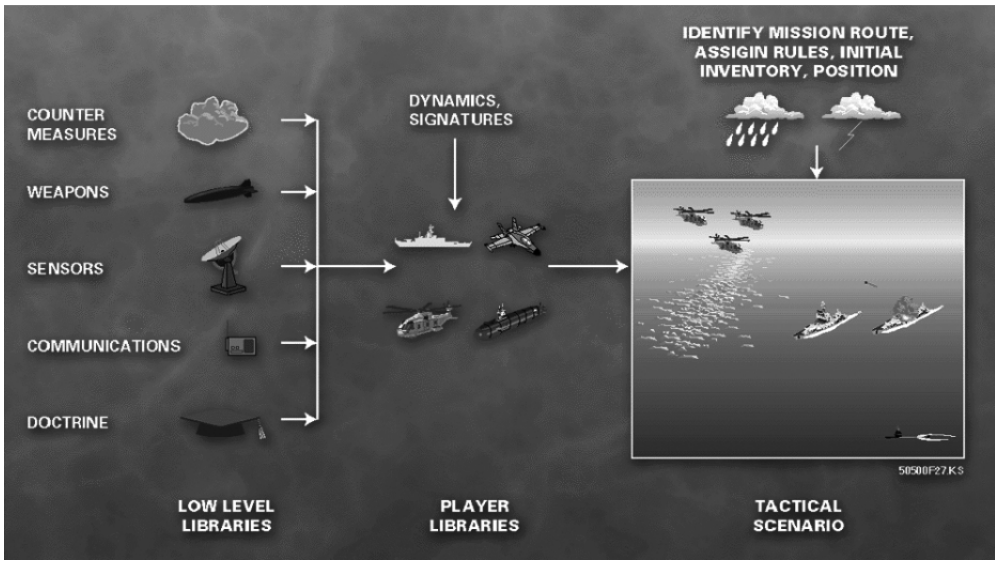


Figure 3. ITEMS Architecture

b) Air Target Definition

Typical parameters include lift and drag coefficients, engine thrust characteristics, mass and wing area.

c) Rules Database

Each player in a scenario is assigned a rule set. Usually there are two: a prime opponent selection rule set and a maneuver rule set. The opponent selection rule set contains the rules for selecting an opponent, e.g., should fast moving jets take priority over slower moving transport aircraft. The maneuver rule set is the decision kernel that invokes the maneuvers from within the air combat maneuvering database mentioned above. It is the rules defined here that instruct the air target to maneuver either offensively or defensively. The logic to transition from one maneuver

to another is also part of this rule base, i.e., when to switch from pursuit to gun track and finally disengagement.

d) Air Combat Maneuvering Database

An air combat database comprises a series of maneuver routines that can be invoked by the rule base. The rule base tells the system which maneuver to perform and the air combat database carries out that maneuver. Supported are upwards of fifteen different maneuvers to include High "G" Barrel Roll, Immelmann, Pure Pursuit, Gun Tracking, Lift Vector Turn, Scissors, Jinks, Break, High Yoyo, and Low Yoyo. These maneuvers cover the complete spectrum of offensive maneuvering, defensive maneuvering and stalemate.

e) Weapons and Sensors Database

The weapons and sensors database is used for specifying the characteristics of the weapons and sensors. Physical models consider the principal factors that affect performance. For example, in the case of the missile the user would be requested to specify the mass and aerodynamic characteristics together with the guidance mode. Countermeasures are defined in a similar manner with specification of radar cross section in the case of chaff and intensity in the case of IR flares.

At scenario load time, the IAT parameters are loaded into memory. At run-time, the aerodynamic coefficients and engine thrust rating data are interpolated based on aircraft state.

The motion path and attitude of the IAT is derived from the applied forces and moments illustrated in Figure 4, which appear as parameters in the Euler equations of motion. As shown in the figure, there are contributions from the effects of gravity, aerodynamic forces and the propulsion system. The axis systems employed include the stability axis, in which all aerodynamic forces are calculated, the body axis and the inertial axis.

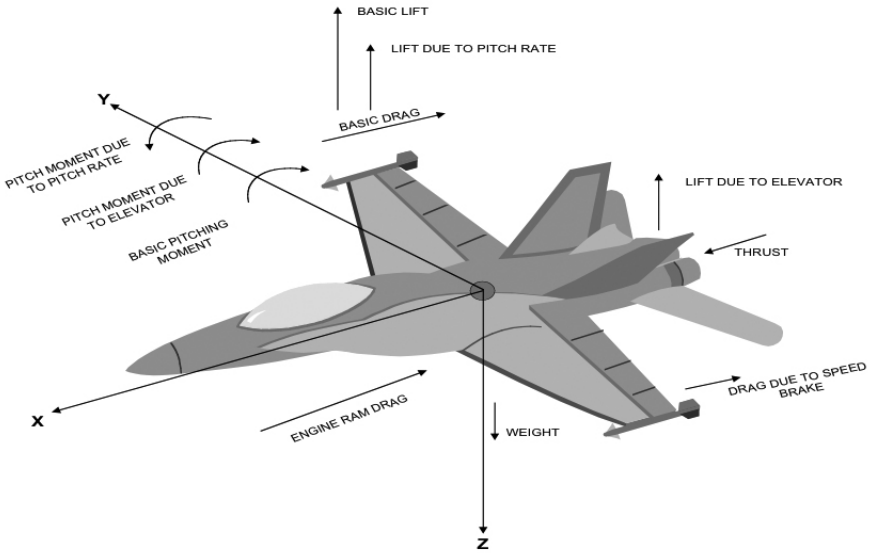


Figure 4. Aircraft Applied Forces and Moments

8. Air Target Maneuvers

Initially, the air combat rules system selects a maneuver from the maneuver list. When the selection is made, the maneuver routine proceeds to compute the position of the control surfaces and throttle position. The resulting aerodynamic loads and thrust are used to maneuver the aircraft to the desired trajectory, resulting in such maneuvers as the pursuit, yoyo, and hard turn.

The control parameters are used by the rules system to select a maneuver from the maneuver list. The list of the condition parameters is unlimited, as the user can always create new parameters as needed. The most commonly used condition parameters are those that relate to the Prime Opponent (PO), typically: Slant range, Angle Off Tail (AOT), Heading Crossing Angle (HCA), closure rate and relative height.

9. Doctrine for Air Combat

The creation of lifelike player behaviour and reaction requires the modelling of player intelligence and this, in turn, is based upon knowledge of player tactics (military doctrine). Tactics, whether used in air, ground or naval applications, require a specialized range of expertise.

Within the SE expert system, each instance of tactical knowledge is represented by IF/THEN rules and is called a “doctrine”. This knowledge provides control over the actions of individual players as well as over the summary actions of groups, such as a change of formation. Doctrines, like player data, are organized into libraries within the DBMS and are referenced by players within the scenario. Examples of doctrines include:

(1) Mission Doctrine: Knowledge pertaining to the mission of a player (goals, routes, contingencies, etc.).

(2) Prime Opponent Selection Doctrine: Criteria for the selection of an opponent for the player in question.

(3) Air Combat Doctrine: Pilot level knowledge controlling the selection of manoeuvres and weapons during air combat.

(4) Command and Control Doctrines: Doctrines applied to players organized into command structures such as flights and strike packages. These doctrines include coordination and control functions such as the cooperation of players in a package towards a common goal and the control/change of formations based on situation.

10. Proficiency

The instructor facility provides for a selection of proficiency levels that give the pilot/aircraft entity various levels of performance when maneuvering. This particular system supports Expert, Intermediate and Novice. Expert aircraft will fly at the sustainable turn and use excess energy in the vertical plane. Novice will not maintain its altitude in a high "G" turn. The Intermediate level is between the Expert and the Novice. Proficiency is built into the maneuvers and the rules sets.

11. Weapons Handling

Weapons handling for the IAT is dominated by the rules. The rules select the weapon type based on the current aircraft state. If the rule is satisfied, the weapon will be selected and the weapon fire request is granted. The IAT aiming module computes the aiming solution angles for the selected weapon. The aircraft is commanded to fly the profile to achieve an aiming solution and to deploy the weapon, either the bombing maneuver for the bombing run or the aiming maneuver for the gun/rockets.

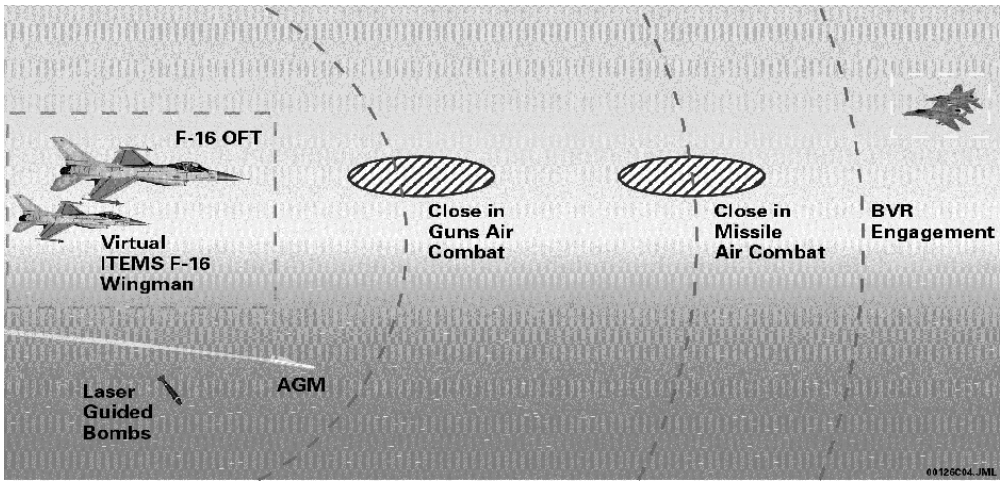


Figure 5. Elements of a Two-vs-Two Air-to-Air Engagement

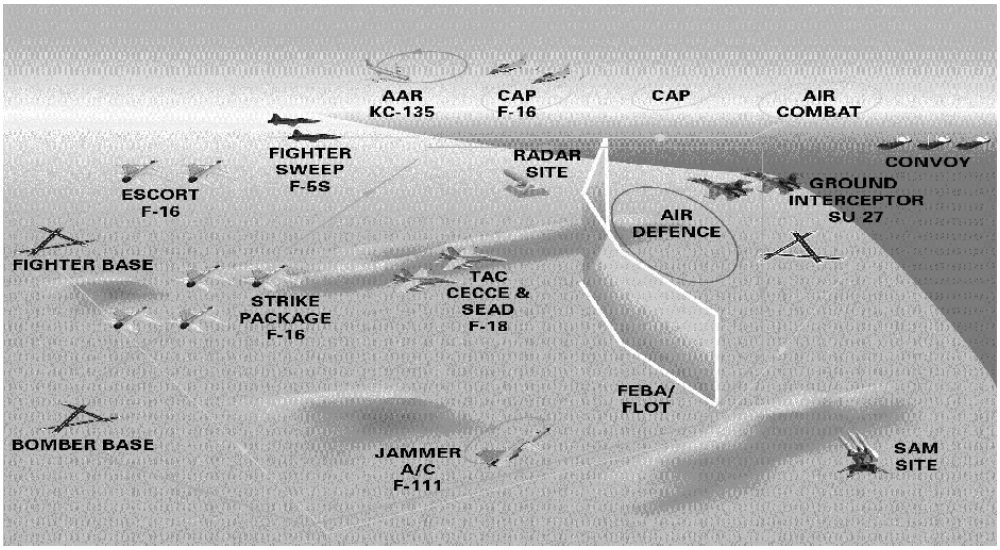


Figure 6. Elements of a Full-up Air Mission

12. Synthetic Environment Application Example 2

Added Complexity—Naval Synthetic Environments

Modeling Issues

Of primary importance is the integration of models within the tactical environment. This implies both correct tactical behavior of the model (i.e. Decision Making) and correct physical behavior.

Tactical behavior issues are generally very user specific. To this end, a solution that allows users to design their own tactics and tactical maneuvers is required. Design cycles with subject matter experts (SME) integrate the complexity and variability required by the user. A hybrid approach permits customized designs for each maneuver and also allows a rule-based definition to be used for any cursory tactics. This approach permits the system to remain flexible in case modified tactics are desired. The ability to inject (instructor) role-play into the system at any time also enhances the flexibility of the system.

Physical model behavior must be comparable to the behavior of systems on the manned simulator in order to avoid introducing dissonant effects that would lead to negative training. Such effects may include:

- Sensor (radar, sonar) differences based upon differing models used for CGF and simulator devices. This could result in detections by the CGF entities that are not consistent with those made by the trainee.
- Navigation differences based upon differing models used for CGF and simulator devices. This could result in CGF entities flying at different altitudes and speeds than the trainee.

To address these issues, models are built for *"closely coupled"* operation with the simulator. The models are either based upon the simulator model itself or are designed specifically to address the CGF model limitations. Within the simulation, model activation may be based upon proximity or some other appropriate measure.

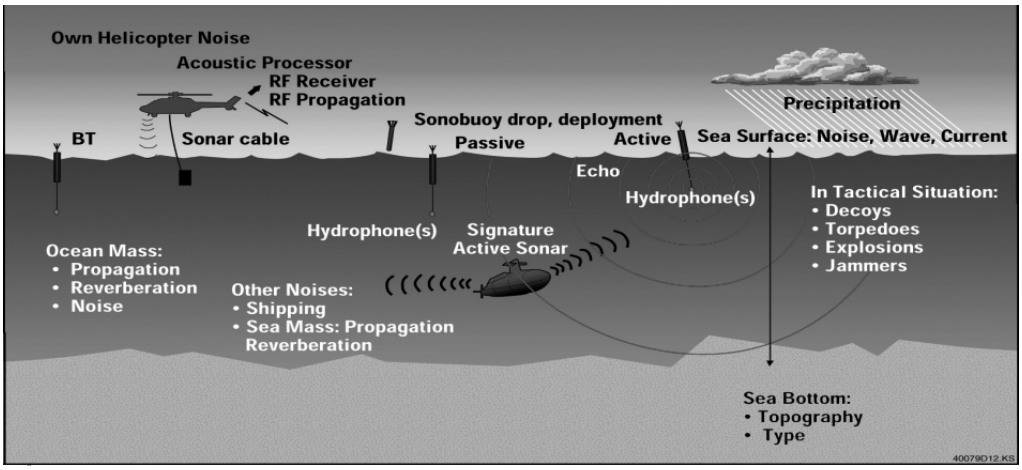


Figure 7. Acoustic Elements in a Naval SE.

13. Coordinated CGF Maneuvers

Coordinated CGF maneuvers represent a level of complexity higher than that for individual maneuvers. As such, they are both more complex in structure and present a more representative and challenging scenario for trainee interaction.

The coordination of entities within a requested tasking implies several units working together towards a single goal. Helicopters may be directed in a plan to search out a reported submarine. Ship groups may be engaged in a datum search or in tactical transit. Several unit types (e.g.: helicopter and ship) may combine in a coordinated tactic of submarine prosecution. A representation of the maneuver categories built could include:

- Helicopter group tactical searches:
  - Bearing and pattern searches for specified contacts.
- Helicopter and ship group tactical searches with target prosecution:
  - Bearing and sector searches for specified contacts. Target prosecution may be role-played or automated via entity rules.
- Helicopter and ship screening maneuvers:
  - Sector patrols with respect to a high value unit. Helicopter dip maneuvers.
- Missile attack reaction strategy:
  - Missile attack reactions for helicopters and ships

14. Coordinated Modeling Issues

While the same technical issues that were pertinent to individual maneuvers are still valid for the coordinated tactics, several new concerns are raised which are specific to the latter.

Primary to these is the method used for coordinating the maneuvers. This includes both sensor coordination and event coordination. Sensor coordination is important in order to ensure that all entities taking part in the maneuver share the same sensor picture. If they did not, synchronized approaches and targeting would not be possible. Similar physical sensor models attached to all CGF entities is one method of maintaining sensor correlation.

A second approach is via Link 11 and contact report message simulation ensuring that tactical contact data is distributed to all members of a communications network. In cases where the maneuver is entirely interface driven, user specified entries might be used.

Event or timing coordination must guarantee that all entities involved are aware of their role (what they must do and when). In order to ensure that entities can choreograph themselves as appropriate within the maneuver, specific events may be generated. These are either provided by the rule-based behavior model or by a state machine based representation of the maneuver progression.

Validation of entity capabilities either at the interface level or at entity definition further ensures that those entities nominated for a maneuver are capable of performing it.

Since the set of maneuvers discussed incorporates multiple entities, one or several simulators may participate by engaging any of the entity positions. In a four aircraft search deployment, for example, simulators may occupy any or all of the four positions, including lead. Additionally, the coordination methodology must support one or more aircraft positions that react independently, that do not provide all the event queues expected by the software and that may not respond as predicted. To support this level of flexibility and interoperability, the maneuver schedulers must have an open concept, as do the control interfaces.

15. Ongoing Developments: Adoption of the High Level Architecture

The High Level Architecture (HLA) is a communication protocol which offers a means to expand the level of fidelity associated with networked devices. The actual content of the information to be exchanged is left to the user(s) when creating the federation. This means that the level of fidelity associated with the data exchange need not be constrained by the standard or format. In future training systems, non time-critical systems such as RWR(defn?), radar and sonar are likely to be networked using HLA.

16. Ongoing Developments: Object Oriented Software Design

The emergence of software technologies such as Object Oriented design will affect what can be achieved by way of training capability as well as how future systems are built and used [1]. Object oriented (OO) design presents a completely new paradigm in building CGF environments. With it come such improvements as greater versatility in creating scenarios, scenario elements and doctrines. As an example, doctrines built in the OO environment are more easily capable of referencing any entity of interest in the scenario. Through plugins, new elements may be introduced into the scenario in a flexible manner without having to rely on pre-defined setups.

The concept of frameworks [4] is also being used to create a more modular open architecture, and one that allows components such as the knowledge-based system to be replaced without re-working the rest of the software. The effects of such advancement on the training arena are yet to be seen but are likely to imply increased realism of simulated systems and an increased coupling between the trainee man-in-the-loop and the electronic battlefield. The trend towards team-oriented training is also likely to continue and to increase in terms of its scope and its importance to the training community.

Some of the next generation software architectures are using the HLA concepts as an integral part of the design, for example OneSaf and STRIVE [5]. These new architectures readily lend themselves to HLA networking without the need for wrappers, and they maximize features of HLA such as time management. In addition, emerging standards such as the Real-time Platform Reference Federation Object Model (RPR-FOM) are likely to aid the simulation model development process by defining the model attributes that must be supported.

17. General Issues and Challenges

The elements of a human-in-the-loop simulation in SE must be integrated in such a fashion that they have, and act upon "knowledge" of each other:

- SE entities must have terrain knowledge and be able to reason with that knowledge
- SE entities must have knowledge about cultural features (e.g. Trees) and virtual players and be able to reason with that knowledge
- SE entities must display realistic performance (no 4000 knot fighter aero models or fully loaded infantrymen leaping tall buildings in a single bound) and must behave (make decisions and respond to the SE/virtual players) in a manner appropriate to their identity (e.g. combat ready fighter pilot, civilian non-combatant in the wrong place at the wrong time) and nominal level of expertise.

SE levels of control can be stacked hierarchically:

- Local command language—sensor/motor loops
- Doctrine/tactics command language—tactical behavior
- Doctrine/C3I command language—strategic behavior

Resolution and fidelity of SE model/database elements can vary substantially—e.g. in an acoustic model:

- Simple—only range may be modeled
- Moderate—include frequency, amplitude
- Complex—include reflections, Doppler effects, etc.

If physical and environmental models in SE systems are compose able, attributes may easily be added or enhanced.

Machine performance and behavior with a human-in-the-loop representation is easier to model and implemented more effectively than human entity representation per SE.

Time critical data must be processed separately from "informational" data.

Level Of Detail (LOD) of physical models and articulated elements must be integrated with knowledge of terrain and environmental conditions such that motion, particularly motion of human entities, is smooth and uninterrupted.

Visual database/texture resolution needs to be matched to entity/cultural feature Level Of Detail to reduce likelihood of visual artifacts such as "skating" and interrupted motion.

The adequacy of visual representation is less and less dependant upon graphics processing power as the technology continues to advance in accordance with Moore's Law.

The realism of visual models in a simulation (level of detail as well as motion fidelity) remains a challenge in realistic articulation of elements such as vehicle antennae and turrets, or, with human figures; posture, joint articulation, response to uneven ground—and is essentially dependant upon the quality and resolution of the physical models driving those entities.

A continuing challenge in both entity and physical models is the representation of multiple similar models (e.g. an infantry platoon), their lawful interactions, and the display of apparently independent behavior.

References

1. Gamma, Helm, Johnson, Vlissides: Design Patterns, Elements of Reusable Object-Oriented Software. Addison-Wesley Publishing Company.
2. European Computer Manufacturer's Association (ECMA): "Reference Model for Frameworks of Software Engineering Environments". TR/55, 2nd edition, December 1991.
3. IEEE P1516.1 Draft 1: High Level Architecture (HLA) - Federate Interface Specification. IEEE, April 1998.
4. Siksik, D., Beauchesne, G., Holmes, R., "The Integration of Distributed Interactive Simulation Protocol Within an Interactive Tactical Environment". Royal Aeronautical Society, London, 1994.
5. Howells, P.B., Charbonneau, M., Kwan, B., "Simulation of Fixed-Wing Air-to-Air Combat using ITEMS". AIAA Conference, New Orleans, August 1997.
6. Simon, E., Howells, P.B., "Experiences in Creating and Managing Networked Simulations". Networking in Simulation and Training Proceedings Royal Aeronautical Society, London, November 1998.
7. Howells, P.B., Simon, E., "On the Use of Frameworks for Real-Time Simulation Applications". Proceedings SIW Workshop Fall 1999.
8. Valade, S., Howells P.B., Simon, E., Gagnon, F., "On the Development of a Synthetic Tactical Real-Time Interactive Virtual Environment". Proceedings SIW Workshop Spring 2000. Spring 2000 SIW conference.
9. Enumeration and Bit Encoded Values for Use with Protocols for Distributed Interactive Simulation (DIS) Applications accompanying IEEE Stds 1278.1-1995 and 1278.1A-1998

Chapter 4
Telerobotic Surgery: An Intelligent Systems Approach to Mitigate the Adverse Effects of Communication Delay

***

Frank M. Cardullo
Dept. of Mechanical Engineering, State University of New York
Binghamton, New York, U.S.A.

Harold W. Lewis, III
Dept. of Systems Science, State University of New York
Binghamton, New York, U.S.A.

Peter B. Panfilov
Dept. of Computing Systems and Networks
Moscow State Institute of Electronics and Mathematics
Moscow, Russia

1. Introduction

The long term objective of this research is to develop a system for remote robotic surgery which will permit surgery between any two places on earth with a patient in one location and the surgeon in another. In fact this surgery could also be performed with a patient on-board a spacecraft. The precise distance from earth, over which our approach is practical has not yet been determined. The major impediment to remote surgery is the effect of telecommunications delay on the surgeon's performance. It has been shown in a myriad of studies of human in the loop systems that system delays lead to degraded operator performance and ultimately unstable systems. To quote Dr. Richard Satava, *"During my development of the initial telesurgery systems through the Advanced Biomedical Technology program at DARPA, there was principal focus on the systems integration but the program was not able to resolve the latency issue This is an area which has had much speculation but little hard data, resulting in the off-hand dismissal of very remote telesurgery."* [1]

Since the delay cannot be eliminated, in order to accomplish this objective, the only solution is to mitigate the effects of the delay on the surgeon performing the operation. We have developed an intelligent systems approach, which is to have the surgeon operate through a simulator running in real-time. The use of a simulator enables the surgeon to operate in a virtual environment free from the impediments of telecommunication delay. The simulator functions as a predictor and periodically the simulator state is corrected with "truth" data (Won Soo Kim pred display ref).

Several aspects of our approach will make use of a variety of forms of intelligent systems as will be explained below. The goal of mitigating the effects of time-delay in a practical way are so challenging that it can be realized only by making the best use of recently developed approaches to machine intelligence.

It is interesting to note that in the late 1980s, after its inception the utilization of laparoscopic cholecystectomy grew rapidly. However, minimally invasive surgery (MIS) for other operations has not experienced the same pattern of growth. According to Ballantyne [2], the reason is that in general laparoscopic procedures are hard to learn, perform and master. This is a consequence of the fact that the camera platform is unstable, the instruments have a restrictive number of degrees of freedom and the imagery presented to the surgeon does not offer sufficient depth information. The solution seems to be at hand with the significant growth of robotic surgery. This is surgery where-in the surgeon operates through a robot. In a sense this robot is a telemanipulator under the control of the surgeon. The robotic system provides a stable video platform, added dexterity and in some cases a stereoscopic view of the surgical field.

Since proximal robotic surgery seems to be maturing the next logical step in surgical care is to extend to remote applications of robotic surgery. That is to say, the surgeon and the operating console are at one location and the robot and patient at another. The idea of remote robotic surgery, or as some refer to it, telesurgery, has been an objective for some time, especially in the military. This advancement is seen by the military as the means by which the next major improvement in battlefield survivability [3]. In addition to the military application, the technology could be useful if an astronaut were to require emergency surgery while on the space station. Furthermore, perhaps the most ubiquitous application will be in civilian medicine. Patients in medically remote areas would have the option of receiving an operation performed by a renowned surgeon even though the surgeon and patient may be thousands of miles apart.

2. The Time Delay Problem in Telerobotic Surgery

The main impediment to the availability of this technology is the communications delay associated with long distance signal transmission. This delay is inevitable and a consequence of the propagation speed of electromagnetic radiation. Figure 1 illustrates the signal paths.

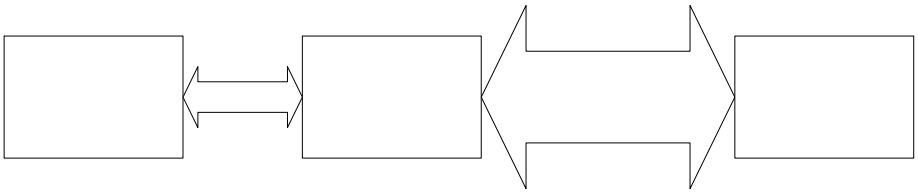


Figure 1. Signal paths.

It is well known that system delays will cause a deterioration of the human-machine system performance. As a matter of fact this is true for any control system, not only a human-in-the-loop control system. Figure 2 illustrates the time domain effect of delays of 0, 200, 400, and 800 ms where the input is a unit step. The graph indicates that as the delay increases, the response lags the input by a greater amount. In addition the 400 ms delay case seems to display limited stability, while the 800 ms delay case clearly exhibits unstable response. The system analyzed includes a fourth order plant and a human operator model, to which the delays are added. Figure 3 presents the results of frequency domain analysis of the same system. Here, one observes that the 400 ms delay case yields a phase margin of approximately zero, while the 800 ms case has a negative phase margin. We can then examine human operator performance

data in a system with and without delays. There are many such examples in the literature. It is observed that when delays become long, human operators will adopt a move and wait strategy. This allows the operator to observe the results of his/her action before committing to another action. The move and wait strategy may be acceptable for controlling a lunar or Mars rover but it is unacceptable in tightly closed loop applications, robotic surgery being one of them.

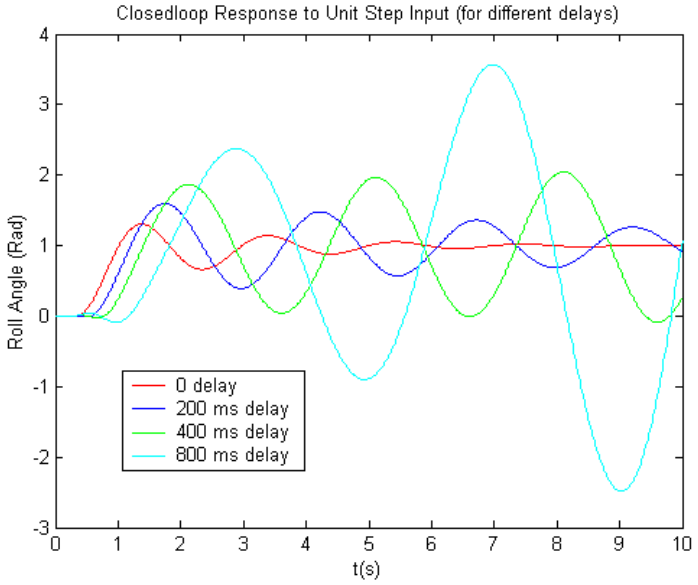


Figure 2. Time domain effect of delays.

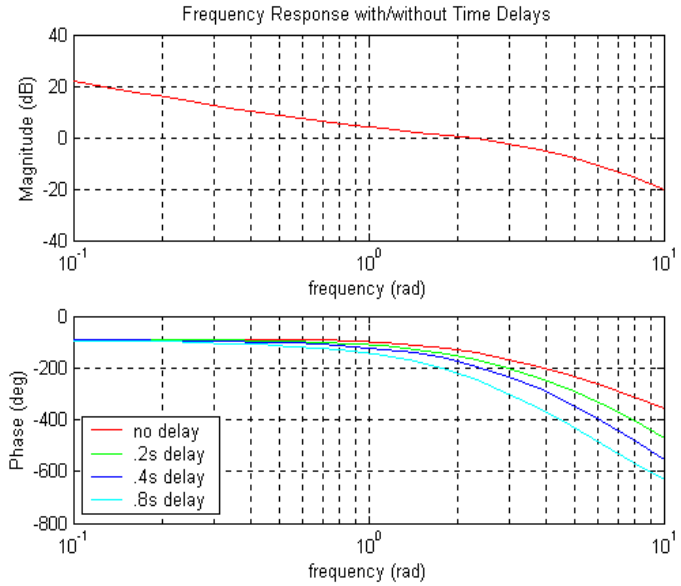


Figure 3. Results of frequency domain analysis of the system.

When the delays in an aircraft flight control system become too long the control loop becomes unstable and the aircraft is said to display pilot induced oscillation (PIO). This is another case where the move and wait strategy will not work. Figure 4 illustrates the effect of delay on a system operator performing a tracking task with and without force feedback. There were several cases of delay (0, 80, 200 and 300 ms) in the force feedback. In all cases the subjects had a narrow field of view visual presentation. The graph shows that at 200 and 300 ms delay in the force feedback the operator's performance is essentially as bad or worse as with no force feedback [4]. Whereas, at a delay of 80 ms his/her performance is much improved and almost as good as a fully synchronous feedback.

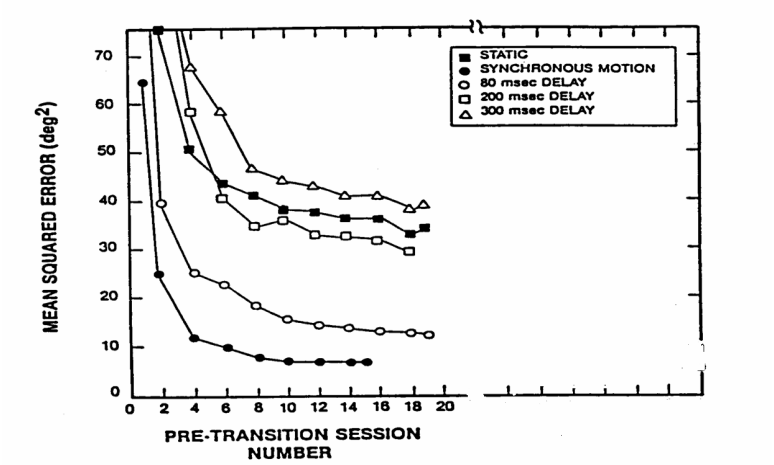


Figure 4. Effect of delay on system operator performance.

3. Preliminary Studies to Determine Maximum Tolerable Delay

Preliminary studies were conducted using experienced laparoscopists in a suture knot tying task. The task was performed using a laparoscopic training device with delays introduced, in 25 ms intervals, into the video monitor via an analog delay device from Prime Image. The performance metric used in this study was the time it took the subjects to complete the knot. Figure 5 illustrates the results. For delays up to about 100 ms the execution time remained relatively constant at about 13 seconds. Above the 100 ms point the time increases substantially. Preliminary results seem to indicate that by the time the transport delay approaches 500 ms the time to complete the knot is about 90 seconds. One of the interesting results is that subjects began to experience nausea at delays approaching one second. This result was quite unexpected based on our considerable experience with simulator sickness.

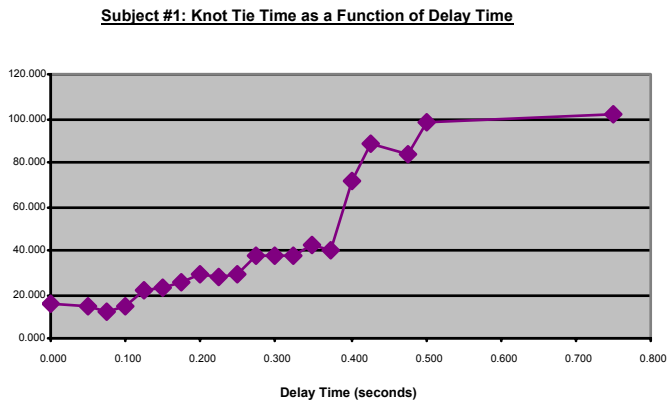


Figure 5. [5] Knot tie time as a function of delay time.

4. The Intelligent Systems Approach

Although universally accepted definitions for such terms as "intelligent system" and "artificial intelligence" may never be found, it is nonetheless useful to start with a reasonable working definition of what we mean by an intelligent system as the term is used in this paper. George Klir, the founding director of the Center for Intelligent Systems at SUNY-Binghamton, states that "Intelligent systems are human-made systems that are capable of achieving complex tasks in a human-like, intelligent way." [6] Restating this slightly, we consider a system to be intelligent if it is capable of a behavior that would be described by a typical observer as an intelligent behavior if a human were to perform it.

Starting from this working definition, we feel that at least four aspects of our scheme, a simplified view of which is shown in Figure 6. for mitigating the time-delay problem in telerobotic surgery are clearly intelligent. First, at least three major components of the approach are intelligent systems when viewed independently. These are the simulator (particularly in its role as a predictor), the image understanding component, and the intelligent controller. Furthermore, the interaction and coordination of all components in the overall integrated system is a complex process that we view as the fourth aspect of intelligence.

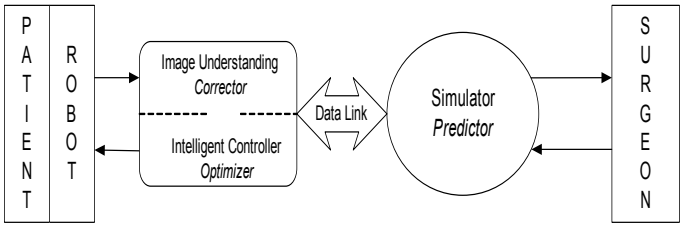


Figure 6. Surgical telerobotics as an intelligent machine.

5. The Simulator as Predictor

Modern simulators tend to be very complex systems in their own right, but one particular aspect of how the simulator will function in this case is where we place the emphasis in calling our simulator an intelligent system. That is, in order for our time-delay-mitigation scheme to work, the simulator must predict what will happen in the surgical field before it happens. Furthermore, the predictive mechanisms in this case are based on dynamic modeling of a far from trivial sort. Adding still more complexity to the task, the system must be designed to allow the models to be updated in real time as the delayed information from the surgical field becomes available.

Clearly, dynamics models both for the robot dynamics and organ dynamics are necessary for the simulator to function in this way. Though both are challenging, the organ dynamics modeling is known in medical research circles to be extraordinarily difficult, particularly in the case of soft tissue. For this, we intend to experiment with a variety of approaches, which will include both finite element analysis and continuum analytical models.

The simulator as used here is clearly a perfect example of an anticipatory system. It is interesting to note that anticipatory behavior is often viewed in the literature [6, 7] as a primary characteristic in intelligent systems.

6. Image Understanding

An image understanding module on the patient side of the communications link (directly connected to the robot) is essential to the functioning of the overall system. The purpose of this component is to recognize the organs and various other objects in the surgical field from the video imagery coming from the surgical camera. It will feed this information on the location and states of these objects both to the intelligent controller (on the patient side) and to the simulator (on the surgeon side). To perform this image understanding task to the level of sophistication required in this research clearly fits any reasonable definition of an intelligent system.

Developing the image understanding module will be a challenging part of the research. One aspect of its design is likely to be an image library. This library will consist of both generic (general anatomical information) and patient-specific (such as from MRI scans) information on the surgical field against which the imagery from the camera will be compared. One aspect of the approach that will facilitate the interpretation of the imagery will be to pick out easily identifiable landmarks as navigational aids.

However, even using the image library and the landmarks, the task of interpreting the input from the camera in real time will still be a very challenging one. Some of the most promising new approaches for real-time image recognition are based on machine learning. In particular, they make use of a fast and very powerful approach called support vector machines, or what are more generically known as kernel-based methods. The power of these methods is that a simple data transformation followed by a linear model effectively constitutes a powerful nonlinear model.

Image understanding encompasses the processing, encoding, and recognition tasks that will be accomplished on the patient side, using the video from the surgical camera(s). The problem of image identification has been a focus of research in the community of image understanding for a long time, and still there is no satisfactory solution available in general. However, when the

problem is more constrained into a specific application domain with a specific application scenario, there are feasible and robust solutions well that are well suited.

A new approach to object recognition and categorization in image is under development by members of our team. A camera image is basically a 2D projection of a 3D world in the field of view of the camera. The approach is to reverse this process by determining the 3D world that generated the image. This approach has been successful in extracting 3D objects, including buildings and trees, from 2D images. The software to do this in real time has been developed to enhance 2D images, detect specified objects, extract 3D objects, and remove others from the 2D image. Although buildings are generally rectilinear, the objects extracted can be of an arbitrary shape. The extraction process is done in two steps. First, the object is detected or recognized. Second, the 3D parameters are extracted.

The method being proposed is geometry-based in that it will use a library of 3D models that can describe what is being seen in the camera (organs, veins, etc.). These models are not rigid models, but rather are models that can change shape depending upon specified parameters. These models are likely to be patient specific, having been generated by an offline process prior to the surgery.

The method decomposes the 2D image into regions. Regions are contiguous sets of pixels that are determined to belong to a specified set based on any arbitrary criteria. Examples of these criteria are color or texture. All objects in the 2D scene are composed of one or more of these regions. The regions in the image are then analyzed to determine the boundaries of the 2D projected objects, with the result being input to algorithms for determining the depth component, defined as the axis normal to the plane of the camera. This component for any surface is determined using calculations based on color, texture, and diffuse and specular reflection from the surface. Some objects can be obscured by other objects. Algorithms are required to determine when this is occurring.

The edges of the 2D objects are found with pattern searches and correlated with the library model data to ascertain the actual 3D structure at the time the 2D image was obtained. The accuracy of the resulting 3D model will depend upon a number of factors, including the resolution of the camera, whether the image is in color or not, the quality of illumination, and the availability of accurate models of the objects being displayed.

7. The Intelligent Controller

As its name implies, we certainly consider the intelligent controller to be an intelligent system, even when viewed as an independent unit. This device, located on the robot/patient side of the communication link, performs in two critical roles. In the ultimate system for use on actual patients, the intelligent controller will be necessary to provide both an added measure of safety and an improved level of efficiency in the presence of time delay. Both the safety role and the efficiency-enhancement role require intelligent behavior.

The need for an added element of safety in the presence of time delay is quite obvious. For a variety of reasons, even when the surgeon as well as the various other components of the system are performing perfectly, the existence of time delay prevents the possibility of 100% certainty as to where various tissue will be in relation to the surgical instruments at any given instant in the

future. Because the intelligent controller will be proximate to (and linked directly to) the robot, it will interact with the robot without significant delay, and thereby has the potential to control all robot movements instantaneously. Thus, as a last line of defense against the possibility of accidental collisions between surgical instruments and the patient's vital organs, the intelligent controller will ultimately play a critical role.

The need for improving the level of efficiency over what it would otherwise be in the presence of time delay is also clear. Finishing surgery in a timely manner and preventing unnecessary frustration for the surgeon are always important goals. While it may be true that the time delays associated with telerobotic surgery will never allow it to be quite as efficient as proximate robotic surgery, the goal at least must be to make it ultimately as efficient as possible.

Although in the course of the research, we will attempt to apply a variety of advanced approaches to machine intelligence in designing effective intelligent controller prototypes, some basic aspects need not be particularly complex. For the safety aspect, a fairly effective controller could be based on nothing more than a three-dimensional geometric model of the surgical field combined with a simple type of production rule system. A typical rule for the case of gall bladder surgery might look roughly like the following:

IF left end-effector holds sharp instrument AND
instrument is within 5 mm of common bile duct AND
an override command has not just been submitted by the surgeon

THEN stop movement of left end-effector immediately,
send safety alarm signal to surgeon,
wait for reset by surgeon

The production rule system for the case of gall bladder surgery may be comprised of perhaps a few dozen such rules. This is a very basic form of the traditional approach to artificial intelligence. Using this as a starting point, we can readily add more sophisticated machine intelligence approaches.

One fairly straightforward addition would be the use of fuzzy logic in the production rules. Fuzzy logic is simply a calculus for representing mathematically the way humans use somewhat vague concepts, such as "very close" or "rapidly", when reasoning about complex systems. For example the rule above could be made more sophisticated by changing the antecedent to the following:

IF left end-effector holds sharp instrument AND
(instrument is very close to common bile duct OR
[instrument is somewhat close to common bile duct AND
left end-effector is moving rapidly]) AND
an override command has not just been submitted by the surgeon

Naturally, when using fuzzy logic, it will also be necessary to represent in the system those numerical values associated with such terms as "very close," "somewhat close," and "rapidly." It is quite simple to represent such terms in a particular context for relative locations, velocities, accelerations, and any other types of variables we may use.

Fuzzy logic is one component of what is now known widely as “the soft computing approach.”. Soft computing (SC) is a term coined by Lotfi Zadeh around 1990 to represent the emerging trend to design complex systems based on hybrids of four component methodologies, each of which had been evolving over the previous three or four decades. These component methodologies of SC are referred to most generically as fuzzy logic (FL), artificial neural networks (ANN), evolutionary computing (EC), and probabilistic reasoning (PR). The key concept of this approach is not just that these four methodologies tend to be powerful in and of themselves, but rather that there tends to be a synergistic effect when two or more of them are combined in appropriate hybrids. The SC approach, also referred to as computational intelligence, may seem to the layman a bit like science fiction, but it is a successful and well established amalgam of methodologies in some fields of engineering and is based ultimately on decades of advanced research.

The success of SC has been demonstrated most graphically in the context of feedback control, particularly in inherently complex control applications. There have been very many citations in the literature of the successful application of SC hybrids, including for example in Lewis [8]. Certainly we will experiment with applying them here as well, and we can already state with confidence that they will be useful in the context of the intelligent controller.

We will also experiment with another approach known in general as optimal control techniques, which use a model reference approach. In this case a model of the entire system; patient, robot and surgeon is employed along with a cost function which will be minimized to determine the coefficients of the parameters in the control laws. This need not be viewed as an entirely separate approach in the sense that optimal control concepts are often part of the SC methodology as well.

8. The Integrated Intelligent System

The fourth intelligent element is actually the total integrated system. The total system behaves as the human surgeon would if there were not a performance encumbering delay. Because the simulator through which the surgeon operates is running in real time the surgeon sees reaction to inceptor movements much more quickly than would be the case if he/she were required to wait while the signals made a complete round trip over the long haul network. Since, in robotic surgery, the surgeon is already in a synthetic environment the introduction of a simulator does not significantly alter the physician’s perceptual stimuli The operating station containing the control inceptors and the visual displays is the same as that used to control the surgical robot in the conventional configuration. In fact because of the addition of haptic stimuli the surgeon’s environment will be more stimulating.

As previously stated in our embodiment the simulator acts as a predictor, providing information to the surgeon consistent with the no delay situation. The image understanding portion is the essential corrector. The intelligent controller is designed as an optimizer. Figure 7 is a detailed representation of the proposed system and the general research areas. The following paragraphs explain the architecture.

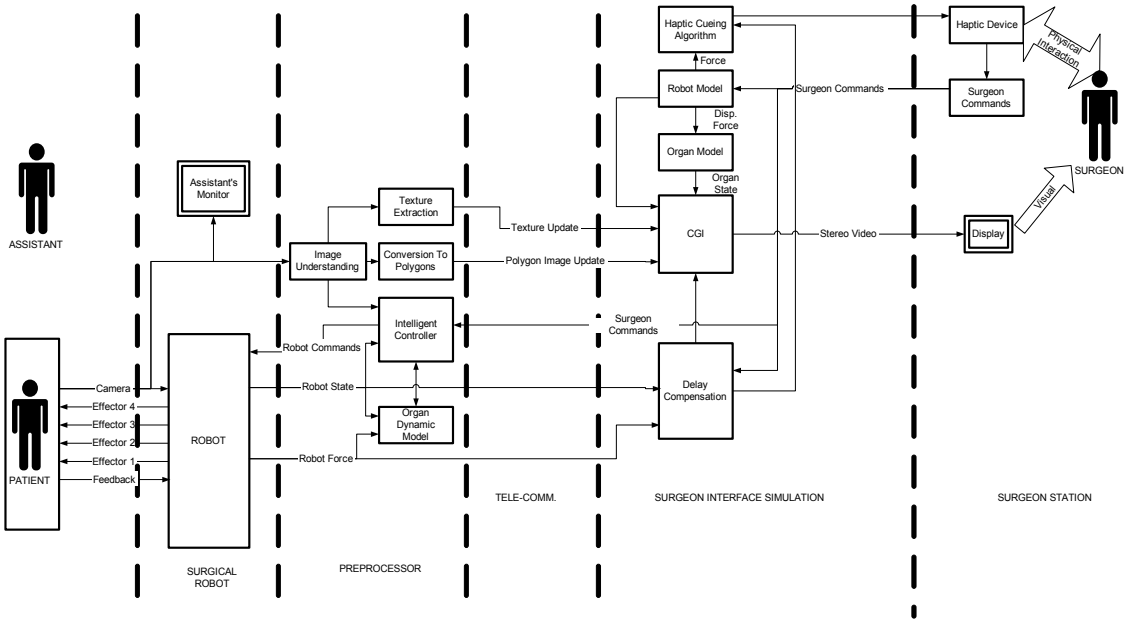


Figure 7. Detailed representation of the proposed system and general research areas.

The signal path from the surgeon's inceptor movement proceeds to the simulator and simultaneously to the intelligent controller, which commands the robot movement. This simulator is like any other in that it calculates all of the system dynamics in real time and from these computations come changes to the system states, which alter the visual scene observed by the physician. The visual scene is generated by high speed computer graphics engines not unlike those employed by modern flight simulators. However, a unique aspect of the proposed embodiment is that the graphics image is periodically updated by the video image transmitted over the long haul network. This approach ensures that the visual scenes at the patient and at the simulator are never allowed to deviate perceptibly. This update is generated by a complex scheme of image decoding, texture extraction and image format transformation.

The intelligent controller performs the dual role of optimizing robot performance and preventing inadvertent incisions. The research will investigate two general approaches to the design. One approach will use optimal control theory and the other will utilize a hybrid of soft computing techniques (fuzzy control, neural networks and genetic algorithms). Both of these techniques have been used successfully to control autonomous aircraft.

The simulator also calculates appropriate inceptor forces. In the near term, the drive signal math model for the haptic stimuli will be essentially the same as that in the actual robot although it will rely on a sophisticated organ dynamics model to compute the appropriate organ forces interacting with the robot end effectors. Eventually haptic feedback will be applied to enhance the environment for the surgeon. It has been shown in other applications that haptic stimuli, even

though artificial, provide information to the operator that improves human performance (reference Hannaford). The organ dynamics model will also provide organ state information to the simulator graphics generator. Another organ dynamics model will reside along with the intelligent controller in order to allow model reference control.

9. Summary

An extremely innovative approach has been presented, which is to have the surgeon operate through a simulator running in real-time enhanced with an intelligent controller component to enhance the safety and efficiency of a remotely conducted operation. The use of a simulator enables the surgeon to operate in a virtual environment free from the impediments of telecommunication delay. The simulator functions as a predictor and periodically the simulator state is corrected with "truth" data.

Three major research areas must be explored in order to ensure achieving the objectives. They are: simulator as predictor, image processing, and intelligent control. Each is equally necessary for success of the project and each of these involves a significant intelligent component in it. These are diverse, interdisciplinary areas of investigation, thereby requiring a highly coordinated effort by all the members of our team, to ensure an integrated system. The following is a brief discussion of those areas.

Simulator as a predictor: The delays encountered in remote robotic surgery will be greater than any encountered in human-machine systems analysis, with the possible exception of remote operations in space. Therefore, novel compensation techniques will be developed. Included will be the development of the real-time simulator, which is at the heart of our approach. The simulator will present real-time, stereoscopic images and artificial haptic stimuli to the surgeon.

Image processing: Because of the delay and the possibility of insufficient bandwidth a high level of novel image processing is necessary. This image processing will include several innovative aspects, including image interpretation, video to graphical conversion, texture extraction, geometric processing, image compression and image generation at the surgeon station.

Intelligent control: Since the approach we propose is in a sense "predictor" based, albeit a very sophisticated predictor, a controller, which not only optimizes end effector trajectory but also avoids error, is essential. We propose to investigate two different approaches to the controller design. One approach employs an optimal controller based on modern control theory; the other one involves soft computing techniques, i.e. fuzzy logic, neural networks, genetic algorithms and hybrids of these.

References

1. Satava, R.M., correspondence with FMC. July 30, 2002.
2. Ballantyne E., Garth H. The Pitfalls of laparoscopic Surgery: Challenges for Robotics and Telerobotic Surgery. Surgical Laparoscopy, Endoscopy & Percutaneous Techniques. 2002; Vol.12, No.1: 1-5.
3. Satava R.M. Surgical Robotics: The Early Chronicles. Surgical Laparoscopy, Endoscopy & Percutaneous Techniques. 2002; Vol.12, No.1: 6-16.
4. Levison W.H., Lancraft R.E., and Junker A.M. Effects of simulator delays on performance and learning in a roll-axis tracking task. Proceedings of the 15th Annual Conference on Manual Control (AFFDL-TR-79-3134). Wright Patterson Air Force Base: Air Force Flight Dynamics Laboratory, 979.
5. Unpublished results
6. Klir, George J., "The Role of Anticipation in Intelligent Systems," in *Computing Anticipatory Systems,* edited by D.M. Dubois, American Institute of Physics, 2002, pp. 37-46.
7. Rosen, Robert, "Anticipatory Systems in Retrospect and Prospect," *General Systems Yearbook,* Vol. 24, No. 11, 1979.
8. Lewis, Harold W., III, *The Foundations of Fuzzy Control,* Plenum, 1997.

Chapter 5
V-Man Generation for 3-D Real Time Animation
***

Jean-Christophe Nebel, Alexander Sibiryakov, and Xiangyang Ju

University of Glasgow, Computing Science Department
17 Lilybank Gardens, G12 8QQ Glasgow United Kingdom
{ jc, sibiryaa, xju}@dcs.gla.ac.uk

Abstract

*The V-Man project has developed an intuitive authoring and intelligent system to create, animate, control and interact in **real-time** with a new generation of 3D virtual characters: **The V-Men.** It combines several innovative algorithms coming from Virtual Reality, Physical Simulation, Computer Vision, Robotics and Artificial Intelligence. Given a high-level task like "walk to that spot" or "get that object", a V-Man generates the complete animation required to accomplish the task. V-Men synthesise motion at runtime according to their environment, their task and their physical parameters, drawing upon its unique set of skills manufactured during the character creation. The key to the system is the automated creation of realistic V-Men, not requiring the expertise of an animator. It is based on real human data captured by 3D static and dynamic body scanners, which is then processed to generate firstly animatable body meshes, secondly 3D garments and finally skinned body meshes.*

1. Introduction

Over the last decade, real-time computer graphics has been subject of vigorous research activity leading to amazing progress and innovation that performance of general-purpose computing platforms has passed the threshold to make it possible to simulate realistic environments and let users interact with these virtual environments with (almost) all senses in real-time. However, creating, animating, and controlling 3D individualized characters is still a long and manual task requiring the skills of experienced modellers and the talent of trained animators on specific software packages- using Maya™, Poser4™ or Reflex-Drama™ - or by scaling generic 3D human models to fit the shape of particular individuals [1] and [2]. Tasks aim at predefining shapes of individuals; deformation and animation are properly incompatible with a really interactive application.

Here we made use of 3D human body scanners to predefine human body shapes. 3D scanning system allowed us to capture whole human body automatically with accurate body shape of individuals and realistic photographic appearances [3, 4, 5, 6]. Directly animating the scanned human body data requires the data to be articulated because there is no semantic information embedded in the data. The articulation can be done manually or semi-automatically [7, 8]. In the conformation approach we proposed, an animatable generic model conformed to the scanned

data that not only articulated the scanned data but also maintained the topology of the generic mesh.

A realistic animation of 3D models requires the knowledge of physical properties regarding the skin and the soft tissues of the real human [9, 10, 11, 12, 13, 14]. In order to collect that information we captured 3D data from people in several key positions. Since we used a scanner based on photogrammetry [15, 16] which has a capture time of few milliseconds, we had the unique opportunity to analyse the skin deformation during the motion between 2 positions. That allowed us to simulate the elasticity of the skin and soft tissues for any vertex of the surface. The association of a skeleton and vertex properties, such as weights, allowed the character skin to be deformed realistically, in real-time, based on correspondence between vertices of the shape and specific bones.

Simulation of cloth in real time is another difficult but possible task if there are static constraints and the detail required is low. Dressing a character is not so simple. This is due to the complicated geometry of a character's body surface and the highly detailed crease patterns that form where clothing is constrained. Simulating the latter patterns in a computationally efficient manner is an open problem but is a necessity to create the detail that essentially characterizes a garment to the human eye. Three main strategies have been used for dressing 3D characters. Garments can be modelled and textured by an animator: this time consuming option is widely spread in particular in the game industry where there are only few characters with strong and distinctive features. Another simple alternative is to map textures on body shape of naked characters; however that technique is only acceptable when characters are supposed to wear tightly fitted garments. In the other approach garments can be assembled: patterns are pulled together and seamed around the body. Then physically based algorithms calculate how garments drape when resting. This accurate and time-consuming technique (requiring seconds [17] or even minutes [18] depending on the required level of accuracy) is often part of a whole clothing package specialised in clothes simulation. Among these strategies, only the third one is appealing since it provides an automated way of generating convincing 3D garments.

Besides conformation, skinning and dressing, a set of skills were built into the characters: there are motion skills, physical characteristics and collision avoidance skills. A V-Man character created is able to adapt his movements and actions to his environment and situation, to walk on any kind of terrain; to go upstairs, downstairs; to calculate paths [19] in order to avoid obstacles. All these interactions will take into account the physical parameters of the V-Man which are added by skill building during character creation. The user can interact with the V-Men that a V-Man understands high-level multimodal commands.

The V-Man system comprises of V-Man character creation and Character Interaction with environment (Table 1) with increasing level of intelligence from top down. All these enabled us to develop our intuitive authoring system allowing any kind of user, without any particular animation skills, to create, animate, control and interact with 3D virtual characters: the V-Men. The system is a stand-alone VR application capable of exporting animation sequences in different standardise formats; it also allows the users to populate their visual simulations or video games with realistic autonomous virtual characters. The V-men characters can be imported into major computer graphics applications (3D Studio, Maya, etc.) and virtual worlds.

Table 1. V-Man System		
V-Man Character	Interaction with Environment	Level of intelligence
• Conformation to articulate the scanned data and to create animatable meshes • Skinning to mimic the deformation due to muscle movements • Animation skills and Physical characteristics • Collision avoidance skills	• Physical animation • Path planning • Get target objects • Voice and keyboard control	Low ↓ High

2. Creation of V-Man Character

V-Man character creation started from an animatable generic model. The generic model conformed to the scanned shapes of individuals to define the shapes of characters the same time maintained the animatable body mesh structure. 3D garments were modelled through the conformation procedure also. In skinning we took advantages of our dynamics scanners to associate vertices with weighted bones so that realistic surface deformation could be achieved. Essential skills were built in the characters for interactive and intelligent animation, such as motion skill, collision avoidance, body size and weight and joint limits.

2.1 Conformation

The conformation algorithm, which conforms a generic model to scanned data, comprises a two-step process: global mapping and local deformation. The global mapping registers and deforms the generic model to the scanned data based on global correspondences, in this case manually defined landmarks. The local deformation reshapes the globally deformed generic model to fit the scanned data by identifying corresponding closest surface points between them and then warps the generic model surface in the direction of the closest surface.

2.1.1. Global Mapping

Global registration and deformation are achieved by means of a 3D mapping based on corresponding points on the generic model and the scanned data. The 3D mapping transforms the manually defined landmark points on the generic model to the exact locations of their counterparts (also defined manually) on the scanned data. All other points on the generic model are interpolated by the 3D mapping; this mapping is established using corresponding feature points through radial basis functions [20].

The global mapping results in the mesh of each body component of the generic model being subject to a rigid body transformation and then a global deformation to become approximately aligned with the scanned data.

2.1.2. Local Deformation

Following global mapping, the generic model is further deformed locally, based on the closest points between the surfaces of the generic model and the scanned data. The polygons of the generic model are displaced towards their closest positions on the surface of the scanned data. Since the body models have been segmented, it is possible to avoid making erroneous closest point associations between points within the surfaces of the limbs and the torso.

An elastic model was introduced to the second step of the conformation. The global deformed mesh is regarded as a set of point masses connected by springs. During the second stage of the conformation process, the mesh anchored to landmarks is relaxes to minimize its internal elastic energy. The displacements of the vertices deformed to their closest scanned surface are constrained to minimize the elastic energy of the mass-spring system. The governing equation to vertex ***i*** is

$$m_i \ddot{X}_i + d_i \dot{X}_i + \sum f_{ij} = f_i^{ext} \qquad (1)$$

where m_i is the mass of the vertex ***i***, d_i its damp factor, f_{ij} is the internal elastic force of the spring connect the vertex ***j*** to ***i*** and $f^{ext}{}_i$ the sum of the external forces.

The relaxation takes the following steps, more details can be found in [0] and [0]:

a. Every vertex deformed to its closest scanned surface except the vertices of the facets intersected the landmarks.

b. The sum of the internal forces on each vertex is calculated, no external force in this model.

c. The acceleration, velocity and position of each vertex are updated.

d. Repeat step a to c until the mesh is settled or the iteration exceeds a fixed number.

Having established global correspondences between the generic model and the 3D captured data, through the landmark-guided articulation procedure, the generic model has been deformed globally to align with the 3D captured data. Exact correspondence between landmarks has been maintained while the positions of all other points have been interpolated. Comparing the 3D imaged model data (Figure 1b) to that of the generic model (Figure 1a), the polygons of the generic mesh were deformed towards their closest positions on the surface of the captured data. Figure 1c shows this final conformation result, rendered using smooth shading. Figure 2 shows the magnified wire-frame body surface representations to illustrate the differences in mesh topology between the captured human body data mesh, Figure 2b, and the conformed generic model mesh, Figure 2c. The conformed generic model has the same mesh topology as the generic model, Figure 2a, but has the individualized shape, i.e. topography, of the 3D imaged real-world data.

Figure 1. (a) Generic model; (b) 3D imaged body data; and, (c) final conformed result.

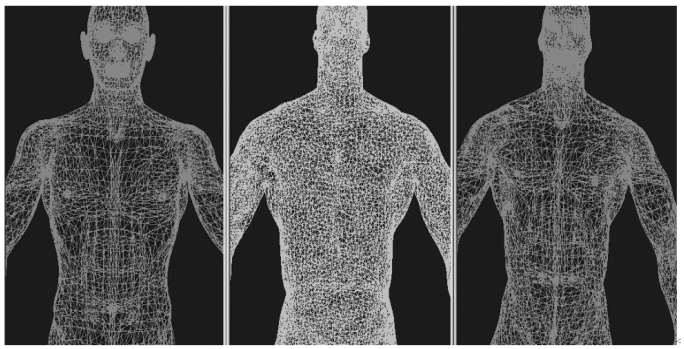


Figure 2. (a) Generic mesh; (b) 3D imaged body mesh; and, (c) final conformed mesh.

2.1.3 Garment

To dressing up the V-Man characters, we offered an innovative technical solution for the generation of 3D garments by capturing a same individual in a specific position with and without clothing. Generic garment meshes are conformed to scans of characters wearing garment (Figure 3) to produce 3D clothes. Then body and garment meshes can be superposed to generate the 3D clothes models.

To dressing up the V-Man characters, we offered an innovative technical solution for the generation of 3D garments by capturing a same individual in a specific position with and without clothing. Generic garment meshes are conformed to scans of characters wearing garment (Figure 3) to produce 3D clothes. Then body and garment meshes can be superposed to generate the 3D clothes models.

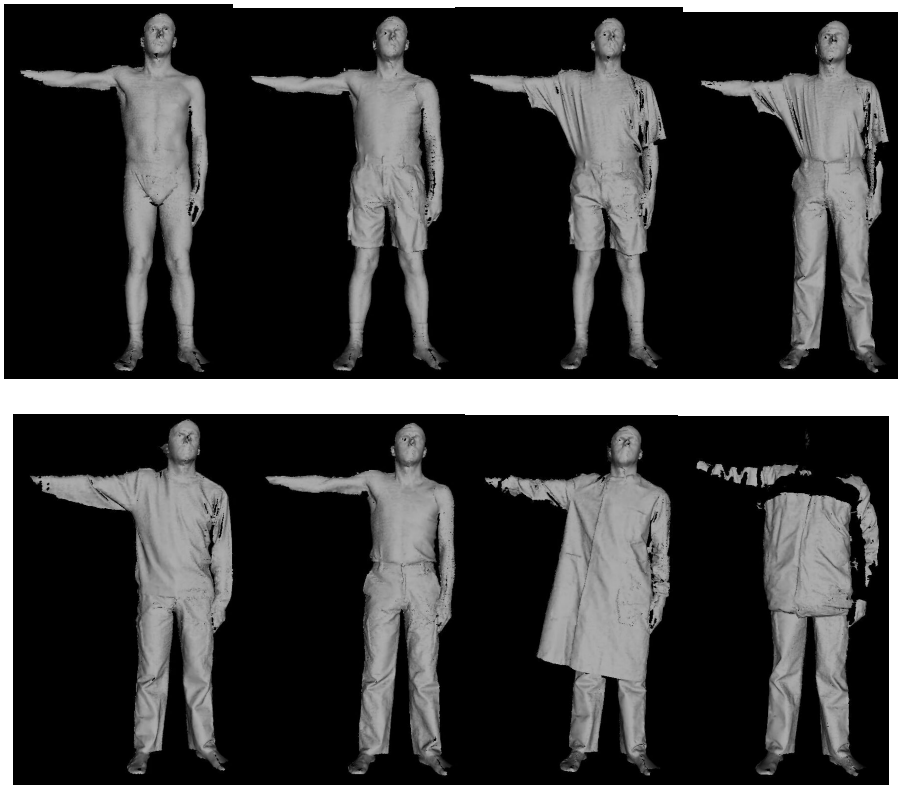


Figure 3. Same male model scanned in different outfits.

Scanning data are conformed using different generic meshes. There is one generic mesh by type of outfit, which provides an automatic garment extraction (Figure 4).

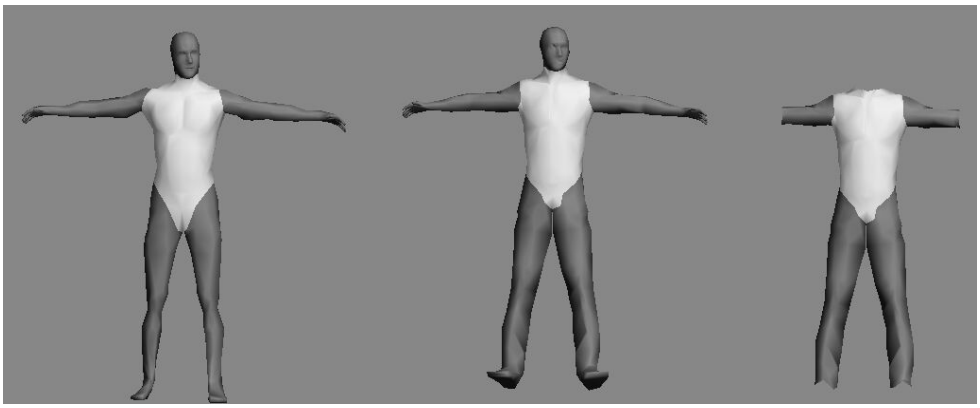


Figure 4. Conformed scanned model without (a) and with (b) garment and garment extraction (c).

The garment mesh is fitted on the naked mesh (Figure 5). Since we work with conformed mesh of generic topology, that fitting process is fully automatic. However, because of the accuracy limitation of the mesh, at that stage it cannot be ensured that the garment mesh will always be above the naked mesh, in particular in areas where clothes are tightly fitted—such as on

shoulders. Therefore a final process detects all triangles from the naked mesh that intersect with the garment mesh and move them backward.

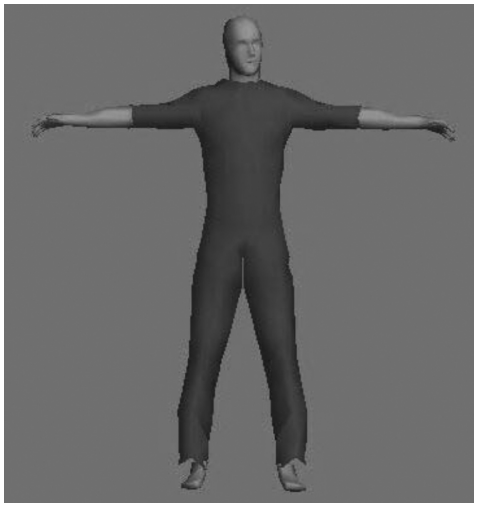


Figure 5. Scan superposition.

Since the number of outfits with different geometries is quite low, the variety of clothes and their important features will come from texture maps. Therefore each outfit is connected to a set of texture maps providing the style of the garment. Moreover users are provided with user-friendly software that allows them to create new texture maps in a few minutes.

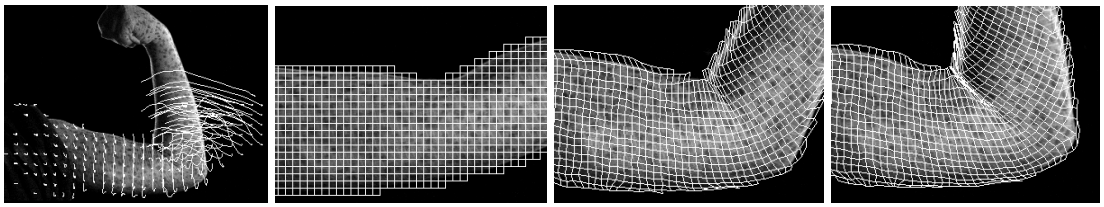
The process of texture generation is the following: using the given flatten mesh of a generic mesh and a set of photos which will be used as texture maps, users set landmarks on the photos corresponding to predefined landmarks on the flatten mesh. Then a warping procedure is operated so that the warped images can be mapped automatically on the 3D mesh (see Figure 5). Moreover areas where angles should be preserved during the warping phase—i.e. seams defining a pocket—can be specified.

2.2 Skin and Soft Tissues

3D character animation in most animation packages is based on the animation of articulated rigid bodies defined by a skeleton. Skeletons are supporting structures for polygonal meshes that represent the outer layer or skin of characters. In order to ensure smooth deformations of the skin around articulations, displacements of vertices must depend on the motion of the different bones of the neighbourhood. The process of associating vertices with weighted bones is called skinning and is an essential step of character animation. Tools are provided by these animation packages usually generate geometric deformation only and it is usually an iterative process to mimic realistic anatomical deformation.

Instead of iteratively trying to converge towards the best skinning compromise by hand, we offer to skin automatically a model from a set of 3D poses which are anatomically meaningful. The technology we use to generate range data is based on stereo-pair images collected by the camera pairs, which are then processed using photogrammetric techniques [0]. By tracking features over

time we could generate optical flows representing the deformation of the human skin. By combining series of range maps with their corresponding optical flows, we can then generate the range flows we need for the analysis of soft tissue deformation. Using the set of 3D points, we can trace each point from the reference posture and obtain the 3D deformation of each point (range flow).

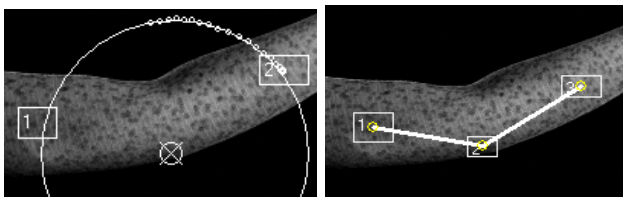


(a) (b) (c) (d)

Figure 6. Using range flows for skinning: (a) point tracing; (b) the reference image with its reference grid; (c) and (d) two images from the sequence with their deformed reference grids.

First we select one reference image and its corresponding 3D model from the sequence. Using the range flow we can trace each point from the reference image and obtain a deformation of any reference grid as shown on Figure 6 for the 2D case. Our method consists in obtaining the joints of the skeleton bones and computing the weights of the points of the 3D model with respect to the bones. The manual step of the method is in the approximate selection of two regions belonging respectively to the parent bone and the child bone. Due to the direct relation between the image pixels and the vertices of the 3D model, this operation can be performed on the reference image. The rest of the process is fully automatic. Using the range flow we obtain the positions of the centre of each region in all the 3D models of the sequence. The centre and the orientation of a global coordinate system are set in the parent bone region and the positions of the centre of the child bone region are then registered in that system. We assume that the bone motion is nearly planar (so we do not consider bone bending). Therefore we can fit a plane passing through the origin of the coordinate system and all the registered positions of the child bone centres. Then we analyse the 2D-motion in that plane. First we project the positions of the child bone centres in that plane, and then since the motion is circular, we can fit a circle on these points. The centre of the circle represents the 2D position of the joint and its 3D position is calculated. In Figure 7a the manually selected regions are shown as rectangles (the first rectangle represents the parent bone region and the second one is the child bone region). The small circles show the positions of the centre of the child bone region in the parent coordinate system. Figure 7a also shows the fitted circle with its centre defining the position of the joint.

Now two 3D vectors connecting the joint and the user defined regions can be fully determined for the whole sequence (Figure 7b). We consider them as virtual bones because they rotate around the joint and coincide more or less with the real bones. With a wider field of view, we could have calculated by the same method the positions of the 3 joints that would have defined more precisely the positions of these virtual bones.



(a) (b)

Figure 7. (a) Joint determination and (b) virtual bones.

The next step of the method is to assign to each vertex of the 3D model a set of weights associated to each bone. We use the following model of vertex motion:

$$\mathbf{x'} = \sum_{i=1}^{n} w_i \mathbf{R}_i \mathbf{x}_i \tag{2}$$

where **x'** is the deformed position of the vertex, n is the number of bones, w_i is the scalar weight associated to the i-th bone, $\mathbf{x}_i$ is the original position of the vertex in the i-th bone coordinate system and $\mathbf{R}_i$ is the transformation matrix of the i-th bone.

The 3D-rotation matrices **R** can be found from the previously calculated motions of the virtual bones. Let $\mathbf{r}_0$ and **r** be the vectors defining a virtual bone in the reference 3D model and in any other 3D model. A bone rotation can be described by an axis **p** and an angle α.

$$\mathbf{p} = \frac{\mathbf{r}_0 \times \mathbf{r}}{|\mathbf{r}_0 \times \mathbf{r}|}, \quad c = \cos\alpha = \frac{\mathbf{r}_0{}^T \mathbf{r}}{|\mathbf{r}_0||\mathbf{r}|}, \quad s = \sin\alpha = \frac{|\mathbf{r}_0 \times \mathbf{r}|}{|\mathbf{r}_0||\mathbf{r}|} \tag{3}$$

and using the Rodrigues formula:

$$\mathbf{R} = \begin{bmatrix} c+(1-c)p_x^2 & (1-c)p_x p_y - sp_z & (1-c)p_x p_z + sp_y \\ (1-c)p_x p_y + sp_z & c+(1-c)p_y^2 & (1-c)p_x p_y - sp_z \\ (1-c)p_z p_x - sp_y & (1-c)p_z p_y + sp_x & c+(1-c)p_z^2 \end{bmatrix} \tag{4}$$

For simplicity we present the 2D case of the motion of a two-bone system around a joint. For this we project the 3D-vertices to the plane previously described. In this case the $\mathbf{R}_i$ are 2D-rotation matrices and equation (1) becomes:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = w_1 \begin{bmatrix} a_1 & -b_1 \\ b_1 & a_1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} + w_2 \begin{bmatrix} a_2 & -b_2 \\ b_2 & a_2 \end{bmatrix} \begin{bmatrix} x_2 \\ y_2 \end{bmatrix} \tag{5}$$

where w1 and w2 can be easily found. Real motions of skin points are more complex then those expressed by the model (1). They are determined not only by the skeleton but also by muscles and soft tissue properties. Therefore the weight values obtained from (3) do not necessary satisfy to the following conditions: $\Sigma\, w_i = 1$, $0 \le w_i \le 1$. To obtain consistent values we normalise and threshold the weights:

$$
\begin{aligned}
& w_1 = w_1 / (w_1 + w_2) \\
& \text{if } w_1 < 0 \text{ then } w_1 = 0 \\
& \text{if } w_1 > 1 \text{ then } w_1 = 1 \\
& w_2 = 1 - w_1
\end{aligned}
\tag{6}
$$

The weights are computed for each vertex of each 3D model generated from the sequence. To obtain a smooth distribution of weights the temporal averaging of the weights of each vertex is used (Figure 8).

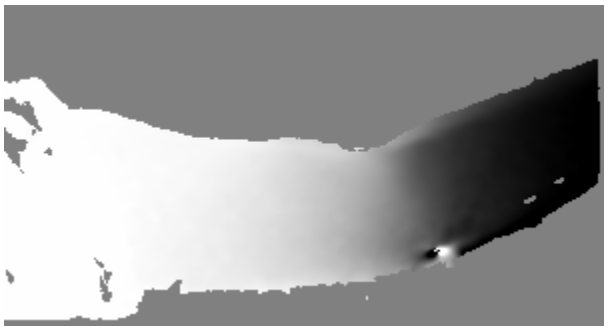


Figure 8. Weight distributed (a white value means $w = 1$ and a black one means $w = 0$).

Using the range flow we obtain the positions of the centre of each limb in all the 3D models of the sequence. The centre and the orientation of a global coordinate system are set in the parent bone region and the positions of the centre of the child bone region are then registered in that system. Then we analyze the motion of each point in its own local coordinate system and assign to each point of the 3D model a set of weights associated to each bone. Finally, once the skinning process is completed, the original 3D mesh can be animated in a realistic way.

3. Skill Building

3.1 Animation Skills and Physical Characteristics

A set of skills were built in character to synthesise motion at runtime depending on their environment, their task and their physical parameters, including motion skills, body sizes and weights and joint limits. Characters adapted skills at runtime to fit specific situations.

Each motion skill is a short sequence of motion capture data defining a single action: real actor motions are transferred and adapted to synthetic characters that might have different limb lengths or body masses. It is also possible to import and use new motion sequences created either with traditional animation software (3DS Max, Softimage, etc.) or with the tools provided with our system (real-time inverse kinematics and footstep control). Therefore authors have the opportunity to extend their library of movements. Transition motion capture data are defined to refine blending between main animations. For example, one could specify that characters should go from a running state to a loitering state through a walking state.

The real time simulation of character motion using physically based algorithm requires the complexity of skeletons to be reduced: for example all the fingers and the wrist are physically animated according to forearm motion. Simplifications can be adjusted to get the appropriate level of detail. Moreover limbs should be assigned bounding volumes (auto-generated from 3D mesh with manual adjustment if required), centres of mass and weights. Finally each joint has a type and angular limits: each join has up to three DOFs limited to believable postures.

3.2 Collision Avoidance Skills

When characters interact in a 3D environment, collisions are highly likely and must often be avoided. We developed a technique which automatically produces realistic collision-free animation of the upper arms. Our method is based on the latest models of collision avoidance provided by neuroscience [0] that allows realistic interpolation of keyframes in a few seconds. Our scheme was validated by comparing computer generated motions with motions performed by a sample of ten humans. These motions were defined by start and final postures and by an obstacle which had to be passed.

The automatic generation of collision-free paths is an active field of investigation which has been addressed in many different ways. For physically based animations collisions are detected and reactions are computed [0, 0]. In robotics many exact solutions exist to produce collision-free motions. However, since the complexity grows exponentially with the number of Degrees of Freedom (DOFs) their use is practically impossible on a simulated human [0]. Hence schemes that are not complete [0] (may fail to find a path when one exists) or probabilistically complete [0] have been developed which makes the task feasible though still time consuming. Other papers deal with achieving collision avoidance when articulated figures animated using inverse kinematics are reaching a goal [0, 0]. Collisions are detected using sensors and response vectors integrated into the inverse kinematics equation system. This process operates incrementally but does not ensure a coherent motion.

Since our application requires the generation of collision-free motions at interactive rates, the respect of the postures defined by the keyframes and the motion to be realistic we decided to explore another path, through the field of neuroscience.

Research by behavioural neuroscientists into the processes by which the central nervous system co-ordinates the large number of degrees of freedom of movement of multi-joint limbs started during the late 60s [0]. However the first papers dealing with obstacle avoidance in the neuroscience literature date from the early 80s, when it was established [0, 0] that path planning involves an intermediate point near the obstacle.

Sabes et al. [0] addressed the issue of how the intermediate point would be chosen. They suggested that properties of the limbs should be taken into account and they looked for a way of expressing the constraint of obstacle avoidance. Hence, they studied the sensitivity of the arm at the closest point of the trajectory; the sensitivity should be minimum with regard to uncertainty or perturbations in the direction of the obstacle. Their sensitivity model is based on the inertial

properties of the arm. The definition of sensitivity they used was proposed by Hogan [0], who expressed the mobility matrix of the arm in end-point coordinates, W( ) as:

$$W(\Theta) = J(\Theta)Y(\Theta)J^{t}(\Theta)$$
$$I^{-1}(\Theta) = Y(\Theta) \qquad (7)$$

where *Y( )* is the mobility matrix of the arm in actuator coordinates. *I( )* is the inertia matrix of the arm in actuator coordinates and *J( )* is the Jacobian.

Since the mobility matrix is symmetric it may be diagonalized by rotating the coordinate axes to coincide with its eigenvectors. It may be represented graphically by an ellipsoid as in Figure 9. The eigenvectors of W have a simple interpretation: the major (minor) eigenvector is the direction along which force perturbations have the largest (smallest) effect. Thus, the sensitivity model predicts that the near points should cluster toward a preferred axis which is the mobility minor axis.

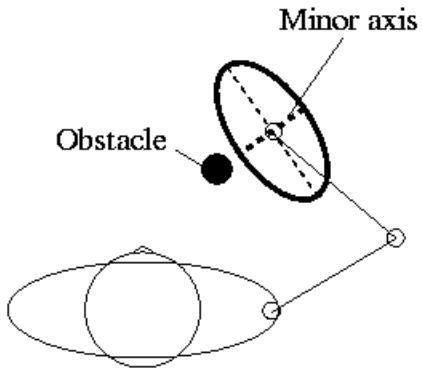


Figure 9. Mobility ellipse in the plane.

Based on Sabes' sensitivity model, our method allows realistic interpolation of keyframes in a few seconds. Keyframes can be created by an animator or selected from previous motion. Once keyframes have been specified, interpolations are achieved to produce the animation. The task we deal with is to offer an interpolation algorithm which generates collision-free motions.

The principle of our scheme is to first detect the objects that should be avoided. An interpolation between keyframes is performed using any classical inbetweening method such as cubic splines. Collisions are sorted and the dominant one is selected for a specific time step. This collision is corrected first. At this time step the frame is modified automatically to generate the intermediate keyframes using geometrical and mobility properties. Finally, this new keyframe is used for a new classical interpolation. This process continues until a collision-free motion is obtained.

This algorithm does not make any hypothesis about the kind of objects which compose the virtual character, only a rigid skeleton is needed. So regardless of the way the articulated figure is defined, our scheme can be used to generate animations without collisions for any collision detection algorithm.

Experimental results comparing a human model with real motions validated our algorithm showing that the generated motions are realistic, simulating the “cautious” (choice of around 30% of participants) way of avoiding obstacles. A more complete description of the process and its validation can be found in the following published paper [0].

4. Interaction Between a V-Man and Its Environment

We developed an innovative technology that enables V-Men to synthesise motion at runtime depending on their environment, their task and their physical parameters. Given a high-level task like “walk to that spot” or “get that object”, a V-Man generates the complete animation required to accomplish the task. A V-Man is able to walk on any kind of terrain, to go upstairs, downstairs, to calculate paths in order to avoid obstacles, and to adapt his movements and actions to his environment. In doing so, the character draws upon its unique set of skills, which are created during the character creation. A V-Man throwing a ball in a pile of cans will generate a realistic animation of the cans. All these interactions take into account the physical parameters of the V-Man. The characters adapts skills at runtime to fit specific situations—for example, the same “grasp” skill can be used to pick up a wide variety of objects, in a multitude of different locations.

The system allows smooth blending between animation sequences and the simulation. The characters can therefore be programmed to take advantage of animation sequences for complex movements while at the same time being sensitive to the physical surroundings in a truly open, interactive way. The overall effect blends the rich, artist-created character of traditional animation with the realism and emergent behaviour of a simulation. The transition between movements required the integration of motion blending algorithms, animation sampling methods and real-time physical simulation of the body.

Path planning, which is of paramount importance for character autonomy [0], was implemented from Luga’s algorithm [0, 0]. This path planner allows a virtual character to autonomously compute, in real time, collision free paths respecting its movement constraints as well as its areas of interest. This algorithm, based on genetic algorithms, is extremely innovative and has been acknowledged as a major step forward by the scientific community.

A V-Man understands high-level commands thanks to a declarative control system. Declarative control is a recent Artificial Intelligence technique that allows the interpretation of high-level multi-modal commands such as “put that there” or “grab this”, where the commands are expressed through out a voice control system while the deictics (“that”, “there”, “this”) are defined by mouse clicks. This technique combines natural language interpretation with a constraint solver [0]. The simplest level of interaction will enable the user to simply define the path for the character to walk along. Beyond that, the characters are provided, through V-Man declarative multi-modal dialogue engine, with declarative commands such as “watch that character”, “follow that character”, or “go there”.

5. Results

Here we present three animations created by our V-Man system to demonstrate the key aspects on interactive animation authoring, physical animation and path planning.

5.1 Demo of interactive Animation Authoring

A female character in the demo picked the white ball on the floor and threw it. A male character walked to a chair to sit down and then walked near the shelf to dance (Figure 10). High level commands such as “pick white ball”, “sit on blue chair”, and “dance” were issued through a keyboard or a microphone. The characters completed the tasks automatically.

Figure 10. Demo of interactive authoring.

5.2 Demo of Physical Animation

In the demo (Figure 11), a man attempted to jump over a wood bar. He collided with the bar and fell. The jump was controlled by motion capture data. When the man collided with the bar, the physical animation started functioning to animate action after the collision. Animations between jump and falling were blended.

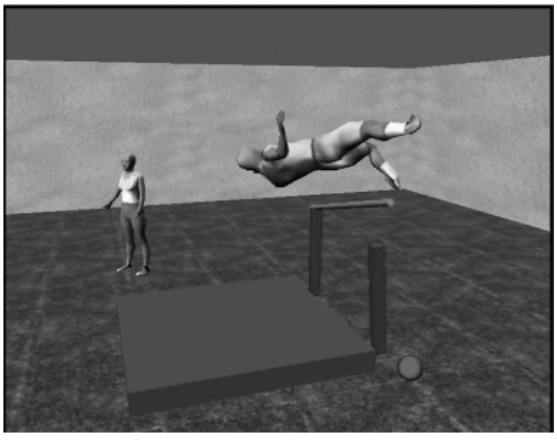


Figure 11. Physical animation and animation blending.

5.3 Demo of Fire Situation

Initially in the demo, people were walking around the building or standing at some positions. After fire alarm (external trigged), each person walked toward his closest exit. Path of individual was planned to avoid obstacles and other people (Figure 12).

Figure 12. Fire in a building.

6. Conclusions

In the paper, we presented our V-man system aimed to create, animate, control and interact in real-time with as minimum user intervention as possible. The key to achieve minimum intervention is the automated creation of V-Man characters. Taking advantages of static and dynamic 3D human body scanner, conformation converted scanned data to an animatable model with an accurate human body and garment shapes; range flows were analyzed for skinning that made surface deformation realistic. In the end, we gave demos to illustration key aspects of our system.

Acknowledgements

We gratefully acknowledge the European Union's Framework 5 IST programme in funding this work. We also want to thank our EU partners who contributed to this work: CS-SI, CSTB, HD Thames, MathEngine, and Sail Labs

References

1. Hilton A., Beresford D., Gentils T., Smith R. and Sun W., “Virtual People: Capturing human models to populate virtual worlds”, CA'99 Conference, 26-29 May 1999, Geneva, Switzerland.
2. D'Apuzzo N., Plänkers R., Gruen A., Fua P. and Thalmann D. “Modeling Human Bodies from Video Sequences”, Proc. Electronic Imaging, San Jose, California, January 1999.
3. Sederberg T. W. and Parry S. R.. “Free Form Deformation of Solid Geometric Models”, Proc. SIGGRAPH'86, pp 151-160.
4. Terzopoulos D. and Metaxas D., “Dynamic 3D Models with Local and Global Deformations: Deformable Superquadrics”, IEEE Transactions on pattern analysis and machine intelligence, Vol. 13, No. 7, 1991.
5. Thalmann N. M. and Thalmann D. “State-of-the-Art in Computer Animation”, ACMCS96, Switzerland, 1996.
6. Thalmann D., Shen J. and Chauvineau E., “Fast Realistic Human Body Deformations for Animation and VR Applications”, Proc. Computer Graphics International, 1996.
7. Nurre J. H. “Locating landmarks on human body scan data”. International conference of recent advances 3D digital imaging and modelling, pp 289-295, 1997, IEEE NJ, USA
8. Ju, X., Werghi, N. and Siebert, J. P., “Automatic Segmentation of 3D Human Body Scans”, IASTED International Conference on Computer Graphics and Imaging 2000 (CGIM 2000), 19-23 Nov. 2000.
9. Daly C. H. “Biomechanical properties of dermis”, the journal of investigative dermatology, Vol. 79, pp 17-20, 1982.
10. Lanir Y. “Skin mechanics”, Chapter 11, in Handbook of Bioengineering, McGraw-Hill, USA, 1987.
11. Behnke A. R. and Wilmore J. H. “Evaluation and regulation of body build and composition”, Prentice-Hall, USA, 1974.
12. measurement of subcutaneous fatness”, Am J Clinical Nutrition, Vol. 32, pp 1734-1740, 1979.
13. Lean M. EJ, Han T. S. and Deurenberg P. “Predicting body composition by densitometry from simple anthropometric measurements”, Am J Clinical Nutrition, Vol. 63, pp 4-14, 1996
14. Han T. S., McNeill G., Seidell J. C. and Lean M. EJ “Predicting intra-abdominal fatness from anthropometric measures: the influence of stature”, Int. J. of Obesity, Vol. 21, 1997
15. Siebert J. P. and Marshall S. J. “Human body 3D imaging by speckle texture projection photogrammetry”, Sensor Review, 20 (3), pp 218-226, 2000.
16. Vareille G. “Full body 3D digitizer”, International Conference of Numerisation 3D - Scanning 2000, 24-25 May 2000, Paris, France.
17. Vasilev T. “Dressing Virtual People”, SCI'2000 conference, Orlando, July 23-26, 2000.
18. Volino P., Thalmann N. M., “Comparing Efficiency of Integration Methods for Cloth Animation”, Proceedings of CGI’01, Hong-Kong, July 2001
19. Luga H., Panatier C., Torguet P., Duthen Y. and Balet O. “Collective Behaviour and Adaptive Interaction in a Distributed Virtual Reality System”, Proceedings of ROMAN’98, IEEE International Conference on Robot and Human Communication, 1998.
20. Ju X and Siebert J. P. “Individualising Human Animation Models”, Proc. Eurographics 2001, Manchester, UK, 2001.
21. Brown J., Sorkin S., Bruyns C., Latombe J. C., Montgomery K. and Stephanides M. “Real-Time Simulation of Deformable Objects: Tools and Application”. Computer Animation, Seoul, Korea, November 2001.

22. Vassilev T., Spanlang B. and Chrysanthou Y. “Efficient Cloth Model and Collision Detection for Dressing Virtual People”, (in CD proc. GeTech Hong Kong), January 2001.
23. Sabes P. N. and Jordan M. I. and Wolpert D. M. “The role of inertial sensitivity in motor planning”, Journal of neuroscience, 1998, 18 (15), 5948-5957.
24. Volino P. and Thalmann N. M. and Shi J. and Thalmann D. “Collision and Self-Collision Detection: Robust and Efficient Techniques for Highly Deformable Surfaces”, Eurographics Workshop on Animation and Simulation, 1995.
25. Bandi S. and Thalmann D. “An Adaptive Spatial Subdivision of the Object Space for Fast Collision of Animated Rigid Bodies”, Eurographics'95, 259-270, 1995.
26. Canny J.F. The complexity of robot motion planning, MIT Press, 1988.
27. Koga Y., Kongo K., Kuffner J. and Latombe J.-C. “Planning motions with intensions”, SIGGRAPH'94, 395-408, 1994.
28. Barraquand J. and Kavraki L. and Latombe J-C. and Li T-Y. and Motwani R. and Raghavan P. “A random sampling scheme for path planning”, Journal of robotics research, 1997, 16 (6), 759-774
29. Zhao J. and Badler N. I. “Interactive body awareness”, Computer-aided design, 1994, 26 (12), 861-867.
30. Huang Z. “Motion control for human animation”, PhD thesis from EPFL-DI-LIG, 1996
31. Bernstein N. The co-ordination and regulation of movements, Oxford: Pergamon Press, 1967
32. Abend W. and Bizzi E. and Morasso P. “Human arm trajectory formation”, Brain, 1982, 105, 331-348.
33. Flash T. and Hogan N. “The coordination of arm movements: an experimentally confirmed mathematical model” , The journal of neuroscience, 1985, 5 (7), 1688-1703
34. Hogan N. “The mechanics of multi-joint posture and movement control”, Biological cybernetics, 1985, 52, 315-331.
35. Nebel J.-C. “Realistic collision avoidance of upper limbs based on neuroscience models”, Computer Graphics Forum, 2000, volume 19(3).
36. Luga H., Balet O., Duthen Y. and Caubet R. “Interacting With Articulated Figures Within The PROVIS Project”, Proceedings of the 11th ACM International Conference on Artificial Intelligence & Expert Systems (AIE), Published in Springer Lecture Notes in Artificial Intelligence, 1998.
37. Panatier C., Sanza C.and Duthen Y. ”Adaptive Entity thanks to Behavioral Prediction”, From Animal to Animates, MIT Press, 2000.
38. Kwaiter G., Gaildrat V., Caubet R., “Controlling objects natural behaviors with a 3D declarative modeller”, Computer Graphics International, CGI'98 Hanover, Germany, 24-26 Jun 1998.

Chapter 6
Interactions with Virtual People: Do Avatars Dream of Digital Sheep?

***

Mel Slater

Department of Computer Science
University College London, Gower Street,
London WC1E 6BT, UK
www.cs.ucl.ac.uk/staff/m.slater

Maria V. Sanchez-Vives

Instituto de Neurociencias
Universidad Miguel Hernandez de Alicante
Alicante, Spain
http://in.umh.es/?page=personalsg&key=50

1. Introduction

In his celebrated book 'Do Androids Dream of Electric Sheep' the author Philip K. Dick explores the relationship between humans and humanoid androids that seem to be human, even superhuman, in every way. The theme is taken up in the film Bladerunner, based on this original story, where the policeman Dekker is required to terminate renegade 'replicants', but falls in love with one himself. In the more recent movie 'A.I.' the question is raised whether you can love a robot (in this case a child) who exhibits every sign of loving you – one slogan of the movie was 'His love is real but he is not'. The book and each of these movies, and many others, explore the moral implications of relationships between humans and machines, machines albeit that are constructed to behave as if human, often also with super-human powers. Their behaviour is so varied, realistic and compelling that the observer is forced to assume that these machines have achieved consciousness, that they know themselves, have emotions and feelings, and know that they have these feelings. Indeed in the case of Dekker there is a tantalizing hint that perhaps he himself is unwittingly a replicant.

These movies and many others paint a popular conception of a world in the not too distant future populated by physically embodied replications of people – robots (entirely non-organic materials) or androids (mixtures of non-organic and organic materials, such as feature in the Terminator movies). The story lines then revolve around relationships between these beings and real humans, whether relationships of exploitation (the replicants in Bladerunner are essentially slaves), proxy love (in A.I. the main artificial character is to be a new son in the family) or war (as in The Terminator). In any case the assumption is that the natural development of artificial intelligence and robotics research will eventually lead to such entities becoming essentially mass produced consumer products available to fulfil a variety of human needs and roles.

This paper explores another form of artificial entity, ones without physical embodiment. We refer to 'virtual characters' as the name for a type of interactive object that have become familiar in computer games and within virtual reality applications. We refer to these as avatars: three-dimensional graphical objects that are in more-or-less human form which can interact with humans. Sometimes such avatars will be representations of real-humans who are interacting together within a shared networked virtual environment, other times the representations will be

of entirely computer generated characters. Unlike other authors, who reserve the term 'agent' for entirely computer generated characters and avatars for virtual embodiments of real people; the same term here is used for both. This is because 'avatars' and 'agents' are on a continuum. The question is where does their 'behaviour' originate? At the extremes the behaviour is either completely computer generated or comes only from tracking of a real person. However, not every aspect of a real person can be tracked – every eyebrow move, every blink, every breath – rather real tracking data would be supplemented by inferred behaviours which are programmed based on the available information as to what the real human is doing and her/his underlying emotional and psychological state. Hence there is always some programmed behaviour – it is only a matter of how much. In any case the same underlying problem remains – how can the human character be portrayed in such a manner that its actions are believable and have an impact on the real people with whom it interacts?

This paper has three main parts. In the first part we will review some evidence that suggests that humans react with appropriate affect in their interactions with virtual human characters, or with other humans who are represented as avatars. This is so in spite of the fact that the representational fidelity is relatively low. Our evidence will be from the realm of psychotherapy, where virtual social situations are created that do test whether people react appropriately within these situations. We will also consider some experiments on face-to-face virtual communications between people in the same shared virtual environments. The second part will try to give some clues about why this might happen, taking into account modern theories of perception from neuroscience. The third part will include some speculations about the future developments of the relationship between people and virtual people. We will suggest that a more likely scenario than the world becoming populated by physically embodied virtual people (robots, androids) is that in the relatively near future we will interact more and more in our everyday lives with virtual people – bank managers, shop assistants, instructors, and so on. What is happening in the movies with computer graphic generated individuals and entire crowds may move into the space of everyday life.

2. Virtual Environment, Immersion and Presence

In most of what follows we will be describing experiments and results that take place within virtual environments (VE) (or 'virtual reality' VR). By a virtual environment we mean a computer generated 'place' in which it is possible for people to interact. There may be events occurring in this place, and there are various forms of interaction. At one extreme the place is static (nothing changes in it) but the human participant can move around within it taking arbitrary positions and orientations. At the other extreme, there may be many events taking place, and the participant is able not only to look at (hear and feel) what is happening but also to intervene and change the course of events.

Virtual environments may be more or less immersive. Immersion breaks down into a number of factors (Slater and Wilbur, 1997):

- Inclusiveness – is the extent to which all sensory data is generated only from within the virtual environment.

- Extensive – is the number of sensory modalities that are accommodated. A system which has vision and sound is more immersive than one that has vision alone.

- Surrounding – is the extent to which the virtual sensory data can be generated from any position and orientation.

- Vividness – the degree of fidelity to every day reality – for example, a system that is able to display shadows in real time is more immersive than one that cannot display shadows.

- Egocentric – information is displayed to the participant to the sense organs in the normal sense of everyday reality. In other words, they see through their own eyes from a first person point of view as if they were there, rather than looking at scenario from the outside (an exocentric point of view).

- Proprioceptive matching – there is a correlation between what they feel as they are moving and what they see, feel and hear as a response. For example, when they feel they are moving their body the sensory response should be appropriate to this – when they turn their head the visual and auditory sense data should match the head turn in exact correlation.

All of the above are ideals, and note that they describe the objective features of a system. Two systems can be at least partially ordered with respect to their degree of immersiveness. Presence, however, is a phenomenon that may arise on the basis of immersion. This is the extent to which individuals respond as if they are in a real world. This response is at multiple levels – low level physiological responses, unconscious behavioural responses, volitional behavioural responses, feelings and emotions, patterns of attention, and so on through to high level cognitive responses – all associated with the feeling of acting in a real place.

In the context of relationships between people and virtual people we can also consider the question of 'presence'. In real life there are typical responses that occur when people interact. For example, eye contact is a particularly important form of interaction which can evoke strong responses – especially if it is held too long. If person A gets too close to person B in a situation which is culturally deemed to be inappropriate (e.g., during a business conversation) it is likely that B will attempt to back away, and feel strong emotions. For people with particular syndromes some types of social interaction can provoke powerful responses. Someone with a fear of public speaking will show strong anxiety responses when forced to be in front of an audience and speak. People with generalized social phobia or shyness will react with strong anxiety to many different types of social situation – such as eating in public, simply attending a party, interactions with members of the opposite sex and so on. People with paranoid delusions will invent entire stories about what is happening around them based on the smallest evidence – a random glance, a coincidental turning away of someone else, two other people who happen to be looking at them while talking amongst themselves, and so on. Confronted with such social situations within a VE, where the other characters are virtual characters, the extent to which these responses are also generated is a sign of presence. In the next section we examine some evidence for this in the context of applications inspired by psychotherapy.

3. Anxiety in Social Situations as a Surrogate for Presence

In this section we consider some examples taken from the realm of psychotherapy. We consider two conditions – social phobia and paranoid ideation. We consider whether people with these conditions will experience similar responses as they would in every day reality. The particular type of social phobia we consider is 'fear of public speaking'. Will people with this condition have the same anxiety speaking to an entirely virtual audience as they would speaking to a real audience? If yes, this is a sign of presence. With paranoia – will people who tend to towards paranoid thoughts (that other people are against them) in everyday life also exhibit such symptoms when the 'other people' are virtual? Again this would be a sign of presence. Of course remember throughout that everyone knows that in fact there are no real people there – so what we are considering are people's automatic responses, not their perhaps higher level thought that 'I know this is not really happening but …' The 'but' is of crucial importance – 'I know this is not really happening but I still feel anxious when those people look at me'.

In this paper we do not at all address the formal scientific results – these are available in other publications which are referenced in the appropriate sections. Here we concentrate only on qualitative aspects of the results.

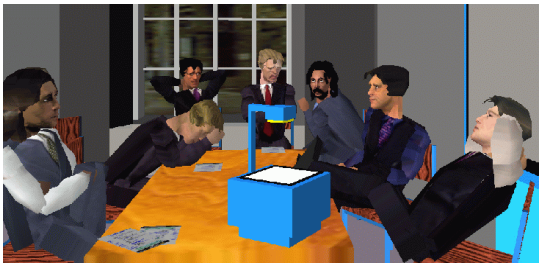
3.1 Fear of Public Speaking

At UCL in 2000-2001 we carried out an experiment in which more than 40 people were exposed to virtual audiences which had three different types of behaviour – they were static (did not move at all), dynamic and showing very positive responses towards the audience, or dynamic and showing very negative responses towards the audience (Pertaub et al., 2001, 2002). Each person experienced only one of these conditions. Some examples of the positive and negative audiences are shown in Figure 1, each consisting of the same eight male virtual characters who changed posture and facial expression, and also made verbal comments during the progress of the 5 minute talk.

Each person experienced the audience using a head-tracked stereo head-mounted display, as shown in Figure 2. A remote operator unseen by the subjects signalled the sequence of responses of the virtual audience, only choosing the timing of each next response, to ensure that each person saw the same thing.

The statistical results indicated that for those who saw the positive or static audience their reported anxiety provoked as a result of the talk correlated with their usual anxiety in everyday life with respect to fear of public speaking. However, irrespective of everyday life anxiety in relation to public speaking the general trend for those who experienced the negative audience was a very strong anxiety reaction. The experimenters noted anecdotally that such subjects had changes in body posture and skin colour, and overall demeanour after experiencing the negative audience indicating a strong negative reaction. (For ethical reasons each person who was assigned to the negative audience group experienced also the positive group before they went away).

Positive Audience



Negative Audience

Figure 1. Virtual Audience for the Fear of Public Speaking Experiment

Figure 2. Head-Tracked Stereo Head-mounted display used in Fear of Public Speaking Experiment

Here are some of the comments made by the subjects who experienced the positive audience:

> 'It was clear that the audience was really positive and interested in what I was saying and it made you feel like telling them what you know.'

> 'I felt great. Finally nobody was interrupting me. Being a woman, people keep interrupting you in talks much more… But here I felt people were there to listen to me.'

> 'They were staring at me. They loved you unconditionally, you could say anything, you didn't have to work'.

Here are some responses to the negative audience:

> 'It felt really bad. I couldn't just ignore them. I had to talk to them and tell them to sit up and pay attention. Especially the man on the left who put his head in his hands; I had to ask him to sit up and listen…. I entered a negative feedback loop where I would receive bad responses from the audience and my performance would get even worse…. I was performing really badly and that doesn't normally happen.'

> 'I was upset, really thrown. I totally lost my train of thought. They weren't looking at me and I didn't know what to do. Should I start again? I was very frustrated. I felt I had no connection to them. They weren't looking at me. I just forgot what I was talking about.'

These comments might seem to be unsurprising – especially the reactions to the negative audience – for after all, their behaviour was extremely hostile. However, it is important to remember that *there was no audience there!* The situation was entirely virtual. What happened expressed the power of the virtual reality to evoke a response that was similar to that of reality – a 'presence' response.

In a later study we compared people's responses to a virtual reality public speaking scenario where the audience was dynamically behaving but neither negative nor positive but neutral in its response. Half of the subjects of this study had anxiety in relation to public speaking and the other half were confident public speakers. Moreover, there were two scenarios – an empty virtual seminar room or the seminar room with the neutrally behaving audience. Half of each group were assigned to the empty room and half to the populated room. Our prediction was that those with public speaking anxiety would respond with greater anxiety to the populated room than to the empty room, but for the confident speakers it would not make much difference whether the room was populated or not. These results were observed across a range of subjective measures of anxiety response and also by heart rate patterns (Slater et al., 2004).

3.2 Paranoid Ideation

Paranoid ideation is the typical pattern of thinking displayed in cases of paranoia. It is characterised by suspiciousness and beliefs that one is being followed, plotted against, persecuted, and so on. There are degrees of paranoid tendencies in a population – ranging from none at all, all the way through to psychotic illness. In 2003 together with colleagues at the Institute of Psychiatry, Kings College London, an experiment was carried out that tested whether the range of paranoid thoughts typically present in a normal population (but excluding people with psychosis) could be reproduced within a virtual reality (Freeman et al., 2003). This experiment was carried out in an immersive projection system (sometimes called a 'Cave') illustrated in Figure 3.

Figure 3(a) shows someone standing in a white box. In fact the floor and three walls are projection screens onto which a stereo image is back-projected. The glasses worn by the person have left and right lenses switching on and off in synch with the images displayed on the screens, thus creating a stereo illusion.

Images are displayed in Figure 3(b) resulting in the illusion of a kitchen. The person perceives a 3D stereo scene, and since his head position and orientation is tracked the images on the four screen-walls knit together to form one overall 3D scenario with high immersion. More information about such systems can be found in (Cruz-Neira et al., 1993).

(a) The projectors are turned off in the ReaCTor

(b) The scene shows a kitchen

Figure 3. Trimension ReaCTor – A Cave-like system

The subjects in the paranoia experiment had a simple task to move through a virtual library. The characters in the library would look at them and make some facial expressions, and objectively they maintained a neutral attitude towards the subjects. Illustrations are given in Figure 4.

The statistical results supported the hypothesis that paranoid thoughts were triggered in the virtual reality in correlation with subjects' propensities to experience these in everyday reality. Here we quote some of the remarks made by the subjects in post-experimental interviews.

> "The two people to the left, I didn't like them very much – well, I don't know, maybe because when I entered the room I felt I was being watched and then they started talking about me. The other people were more neutral and more inviting except the guy with the beard."

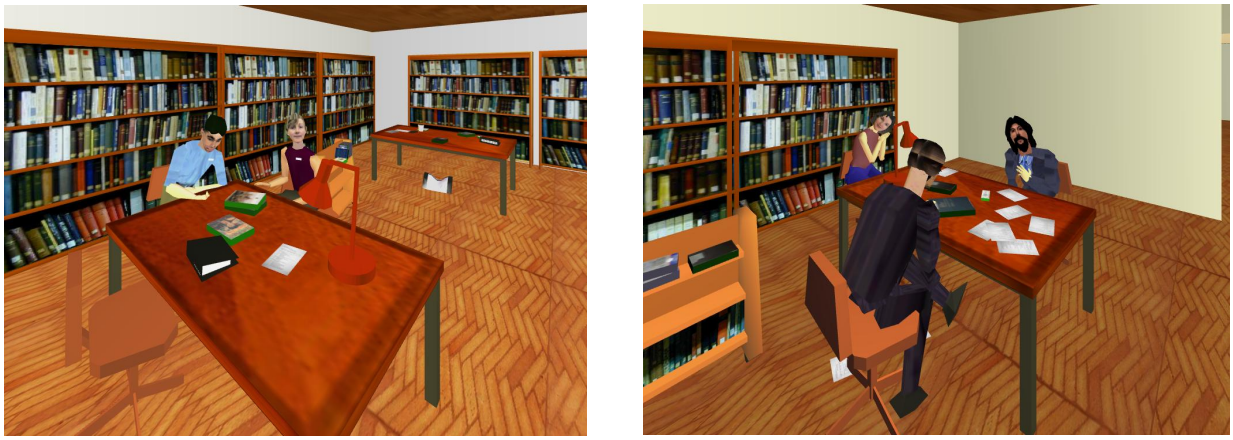


Figure 4. The Scenario for the Paranoia Experiment

> "It was probably more real to me than I expected it to be. At some point, I was trying to navigate around a table and almost found myself saying sorry to the person sitting there. I felt that they were getting annoyed with me for doing that…"
>
> "It was really weird, because they were all definitely in on something and they were all trying to make me nervous. It was clear that they were trying to mock me, they kept on looking at me and when I looked back, they were uuhh… The guy with the suit was really weird because he kept smiling at me and it was quite sinister."

It should be noted that there were no sounds in these environments – so what the subjects heard were entirely made up from within their own minds. The results were quite remarkable – people reacted strongly to the virtual characters, even though objectively everyone knew that there were no people there at all. These studies have been followed up and repeated again, with publications pending.

4. Face-to-Face Meetings Between Real People in a Virtual Reality

Above we considered the results of encounters between real people and virtual people. In this section we briefly consider what happens when real people, who may be physically separated by thousands of kilometres, meet face-to-face in a networked virtual environment. By face-to-face we mean that each person sees a virtual character representing the other one. This virtual character speaks in real-time using the real voice of the remote partner, and at least some of the movements (in particular head movements) of the remote person are reflected in the movements of the virtual character.

An example is shown in Figure 5 – where a person is in the immersive projection system interacting with another who is physically remote. They can talk to one another and have the impression of being in the same extended 3D space which is experienced in stereo. For example, they can make 'eye contact' and (virtually) stand very close to one another if they so wished. Of course the remote person may not be in a projection system – they may be using a head-mounted display or even sitting in front of a conventional monitor, keyboard and mouse system, seeing the virtual scenario on the screen in front of them.

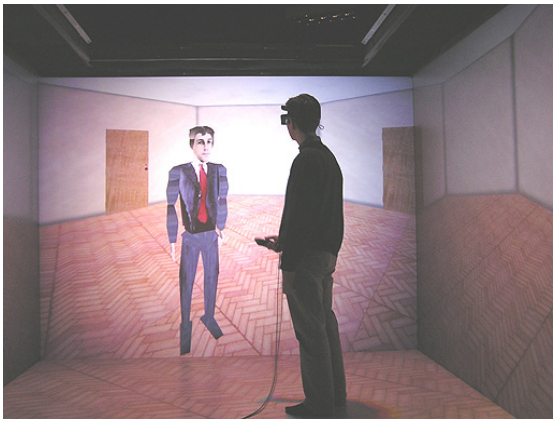


Figure 5. A meeting in virtual reality

Many experiments have been carried out examining two and three-party interactions in such environments (Slater et al., 2000; Schroeder et al., 2001). There are several findings. First greater immersion confers greater social power – typically in a mixed encounter where one person is in a more immersive system than the others (e.g., one is in a head-mounted display and the others on desktop display systems) the one in the more immersive display will have a tendency to become the leader (other things being equal).

The avatar representation must have the capability for the people to express themselves – at the very least make eye contact, make gestures. If these basic features of everyday communication are not possible, then social order tends to break down, with some people even believing that others are deliberately trying to be disruptive and thwart the desires of the others. Similarly problems of network delay so that people for example are unable to immediately hear answers to their questions may become major impediments to smooth social interaction.

The avatar representations must match its corresponding behavioural capabilities. For example, in one experiment one avatar looked very sophisticated (photorealistic) but could do nothing – not move its head nor limbs. The other people represented by simple block-like extremely cartoonish avatars became suspicious of the intentions of this more sophisticated looking person. In another experiment (Garau et al., 2003) people experienced either more sophisticated looking or less sophisticated looking partner avatars, and behavioural capability with respect to eye movement was either more or less in conformance with every day expectations during the flow of a conversation. It was found that consistency between appearance and behaviour was very important to maintain presence between the participants.

In spite of technical problems the possibilities are nevertheless very interesting. Here we have people who are physically in remote places who nevertheless can coexist and interact within a shared 3D space, talk and accomplish physical tasks together. One such experiment concluded that when two remote people carried out a task similar to solving a Rubik cube type puzzle together their performance within two ‘Cave’ systems was almost as good as the performance of people who did the same thing in reality (Steed et al., 2003).

5. Presence and Perception

Above we have informally reviewed some of the evidence demonstrating that people do respond to virtual characters as if they were real. How can this happen? Why do we react to characters that are crude human representations in a way that is similar to how we react to humans? Do emotions overrule our cognitive functions? Understanding these phenomena would take us deep into the workings of the brain and in this section we offer some speculations that may point to where to look for an explanation. First of all, we should consider how we perceive the external world. Perception is constructed from information from different sensory modalities (visual, auditory, tactile, etc.) that arrives to the brain in the form of electrical impulses. For instance, all the information of the visual scene travels codified in a binary code (impulse/no impulse) in a 3 mm wire which is the optic nerve comprised of some one million axons (for each eye). This information is relayed to the thalamic nucleus and from there it will reach the primary visual cortex, to project then to higher or association areas. Different aspects of the visual scene (contours, colours, movement, etc.) will then have to be reconstructed such that they create our internal visual representation of the environment. This processing, from the exterior world all the way up to associative cortices is called bottom-up processing and is driven by so-called feed-forward projections. However, there is evidence to support the view that our perceptual processing does not solely work as a video camera that films and re-plays the environment. One piece of evidence is that the neurons located in the primary visual cortex, and even in the thalamus receive more connections from other neurons in the cerebral cortex (or feed-back connections) than from the sensory organs (Bullier *et al.*, 2001). What this means in functional terms is that the visual reconstruction that these neurons are achieving is deeply influenced by information that already exists in the cerebral cortex, which is called the top-down processing (e.g. Li et al., 2004). What top-down connections convey are the internal factors that affect our sensory perception, including our previous experiences, expectations, motivation, attention, etc. The evidence for the functional correlate of this top-down influence is easily experienced when we look at images like the Kanizsa triangle and similar illusions e.g. (Kojo et al., 1993), confirming the fact that sensory perception is not a passive but an active process where perceiving and interpreting are carried out simultaneously. This implies that our perception is finally determined by the confluence between two streams of information, the bottom-up and the top-down, the external world and the "internal" one.

How does this relate to our perception of avatars? In studies of presence in virtual environments it has been observed that in order to achieve a high level of presence it is not crucial to have a highly realistic visual representation of the environment (Sanchez-Vives and Slater, 2005). A coarse representation may induce a high level of presence; participants in a VE just need to be given some *minimal cues*. The reason why minimal cues can be enough to induce presence is because our perception is an active process and the cortex fills in some of the missing information (Ramachandran and Gregory, 1991; De Weerd et al., 1998). Since our perception is tightly linked to interpretation, the non-sense information is often eliminated and the missing information is filled-in based on previously experienced schemes. One classical example of fill-in processes is the one that takes place in the *blind spot* of our retina. The retina has a discontinuity in the back of the eye at the location were the optic nerve emerges, the optic disc. This area of 1.5 mm diameter lacks photoreceptor cells and should therefore appear as a hole in our perceived visual scene. This does not happen however because that empty space is appropriately filled-in by the brain from surrounding information. In a similar way, probably a

few cues from the avatars such as face elements, eye following, body movements and so on are enticing enough for the brain to perceive this and to react to it as if it were a human.
There is an additional element that makes interaction with avatars special compared to with other elements in the VE: the emotional content. Faces are considered as emotional stimuli. Emotional visual stimuli, are detected faster, they evoke enhanced responses in the visual cortex, and they capture attention more readily than other significant objects in the scene (Lang et al., 1998; Bradley et al., 2003). Face processing occurs even in non-conscious, pre-attentive states, and immediately recalls attention and induces a response. An area in the brain, the fusiform gyrus (inferotemporal cortex)(for a review see Haxby et al., 2000), is specialized in responding to face stimuli. This explains why localized lesions in the cerebral cortex can affect specifically the ability to recognized faces, as it was nicely illustrated in (Sacks, 1998) "The man who mistook his wife for a hat". This area seems to work as well comparing visual inputs with internal representations or recalled images (Frith and Dolan, 1997), or, as we explained above, one of the areas where bottom up and top down information meet each other.

What is so special about faces? Facial expression is one of the elements forming emotion, along with autonomic response (changes in heart rate, respiration, blood pressure, etc.) and with the subjective feelings characteristic with emotion (sadness, fear…). It is also a fast form of non-verbal communication that may have had an important survival role in evolution. For this reason, face recognition is deeply engrained in the brain wiring and even newborn humans express the capability of recognizing schematic representations of faces (Turati et al., 2002). Due to this hard wiring, we willingly identify faces and respond to them, even if they belong to highly schematic virtual characters.

To summarise: top down processing implies that if sufficient minimal cues are provided within a virtual environment, the brain processing fills in missing information. The second element, critical to avatar interaction is that the processing of the human form, especially faces, carries emotional aspects, which may still further minimise the degree of fidelity that needs to be depicted within a VE in order to evoke 'as if it were real' responses.

6. Speculations on the Role of Avatars in the Future

The discussion above was based on a review of scientific work – experiments that were designed the probe the relationships between people and virtual people, and between remotely located real people communicating via avatars, and a brief review of the relevant neuroscience literature about how this process may work. This section is entirely speculative, looking at implications, painting a portrait of future possible proliferation of virtual characters in everyday social life.

We are already familiar with entirely virtual characters in film. For example, in the film Gladiator entire crowd scenes were constructed with computer graphics. In the film Final Fantasy every individual was a computer generated character with almost believable bodies, movements and facial expressions. In films such as Shrek, the characters have excellent postures, gestures, motions and facial expressions, but are mostly humanoid rather than human. In spite of their non-humanness we still laugh at their antics and even maybe identify and sympathise with their plight. The fundamental difference in technology between such movie based virtual characters and those who populate virtual environments is that the movie characters cannot be

generated in real-time (or anything approaching real-time). The rendering and animation requirements are so complex that typically several computers have to work together to produce a frame of animation over several minutes or hours. For a movie every aspect of the rendering and animation has to be as perfect as is possible. Speed is sacrificed to quality. In a virtual reality exactly the opposite must occur – quality is sacrificed for speed, because it is completely impossible to interact with a character which has its display updated once every few seconds or minutes instead of 20 or 30 times a second at the minimum.

However, it is only a question of time. As processing power in general, and in particular the speed and capabilities of graphics processors essentially double once a year, it will soon be possible to populate virtual realities with highly realistic looking and behaving virtual characters – years rather than decades.

Moreover projection and display technology is advancing – albeit at a much slower pace. The idea of Mixed Reality is to bring virtual elements into the realm of everyday life. A 'Cave' is a highly specific example of this – we project onto the walls of a particular built-for-the-purpose room. But the goal is to be able to project anywhere and everywhere, and to blend the virtual into the real in a seamless fashion. 'Presence' still has the same meaning as before – we respond to the virtual as if it were real. However, now the virtual and real form a new totality – neither real nor virtual but something more than the sum of its separate real and virtual parts.

Imagine you walk by a restaurant deciding whether or not to go in. What is important to you is the atmosphere inside – does it seem friendly are there people in there, are they enjoying themselves? You look into one place – and it seems empty – you walk on. You look into another place, and it is full of people. You step inside – and soon come to notice that most of the 'people' in there are in fact virtual characters, programmed to behave as if they were having a great time. Amongst the virtual people are a few real people. Now you can distinguish the difference. You know that in a few years time even this difference will not be noticeable.

Actually the situation is more complex. You notice that at one table there are real people and other virtual people. They all seem to be sitting around the same table and they are conversing. In fact what is happening is that there are real people in this restaurant and real people in another restaurant which might be in another continent, and they are all maintaining the Mixed Reality illusion of sitting down to have dinner together.

Later you need to take a train to get home. You go up to the ticket office inquiry desk because you want to know the most efficient way to get to your home station. The person behind the other side of the glass offering you advice is in fact virtual. The program is sophisticated enough to understand your question and respond with appropriate information. You wonder why anyone has gone to all the trouble to present the information to you in this very sophisticated way, rather than just, for example, on a text display. However, interaction between humans can provide a certain level of reassurance, comfort. After giving you the information the virtual inquiry clerk smiles warmly at you, and wishes you well on your journey. Somehow you cannot help responding in kind, smiling back and wishing the clerk a pleasant evening – even though you know that your reaction is absurd at some level. Absurd but human – it seems that we are programmed to behave in such a way when all the outward signs give the impression that

someone else is there and will respond to you. (The experiments described in the previous section lend support to this supposition).

As you look at the crowds filling up the station concourse, you realize that in fact there are very few real people actually there. There are many reassuring looking people together all having a good time, giving an overall pleasant ambience to this station scene. You're not sure who is real and who is virtual – except that you see a group of children (presumably real) hand in hand with full-sized solid looking characters out of a Disney cartoon.

7. Conclusions

The combination of computer graphics, virtual reality, mixed reality, projection and display technology, artificial intelligence make it more likely that our future will be one where humans interact in their daily lives with virtual characters, rather than the long-predicted time of the coming of the robots. The robots will come but they will be displayed, not physical entities that hurt when they bump into you. As we have seen, being virtual does not reduce the probability that people will react to them with appropriate affective responses.

For people of a certain age today this is already true in a specific sense – people who spend hours playing with computer games already are interacting daily and significantly with virtual characters. People who engage in on-line societies and on-line multi-participant games already partly live in a parallel universe with different normative values, different culture, different objectives.

In time to come such virtual characters individuals and crowds will permeate our society, taking on individual roles as information providers, entertainers, advertisers, advisors, counsellors – and maybe friends, partners, and perhaps – the idea of A.I. – proxy pets and children. At some time in the future we will be in a real scene and no longer know without substantial investigation who is real and who is virtual.

Do avatars dream of digital sheep? Today they don't but we act towards them as if they do. Tomorrow – maybe the answer will be that in fact they do.

Acknowledgments

This work is a contribution to the FET PRESENCIA project.

References

Bradley MM, Sabatinelli D, Lang PJ, Fitzsimmons JR, King W, Desai P (2003) Activation of the visual cortex in motivated attention. Behav Neurosci 117:369-380.

Bullier J, Hupe JM, James AC, Girard P (2001) The role of feedback connections in shaping the responses of visual cortical neurons. Prog Brain Res 134:193-204.

Cruz-Neira C, Sandin DJ, DeFanti TA (1993) Surround-screen projection-based virtual reality: the design and implementation of the CAVE. In: Proceedings of the 20th annual conference on Computer graphics and interactive techniques, pp 135-142: ACM Press.

De Weerd P, Desimone R, Ungerleider LG (1998) Perceptual filling-in: a parametric study. Vision Res 38:2721-2734.

Freeman D, Slater M, Bebbington PE, Garety PA, Kuipers E, Fowler D, Met A, Read CM, Jordan J, Vinayagamoorthy V (2003) Can virtual reality be used to investigate persecutory ideation? J Nerv Ment Dis 191:509-514.

Frith C, Dolan RJ (1997) Brain mechanisms associated with top-down processes in perception. Philos Trans R Soc Lond B Biol Sci 352:1221-1230.

Garau M, Slater M, Vinayagamoorthy V, Brogni A, Steed A, Sasse MA (2003) The impact of avatar realism and eye gaze control on perceived quality of communication in a shared immersive virtual environment. In: CHI '03: Proceedings of the conference on Human factors in computing systems, pp 529--536: ACM Press.

Haxby JV, Hoffman EA, Gobbini MI (2000) The distributed human neural system for face perception. Trends Cogn Sci 4:223-233.

Kojo I, Liinasuo M, Rovamo J (1993) Spatial and Temporal Properties of Illusory Figures. Vision Research 33:897-901.

Lang PJ, Bradley MM, Fitzsimmons JR, Cuthbert BN, Scott JD, Moulder B, Nangia V (1998) Emotional arousal and activation of the visual cortex: an fMRI analysis. Psychophysiology 35:199-210.

Li W, Piech V, Gilbert CD (2004) Perceptual learning and top-down influences in primary visual cortex. Nat Neurosci 7:651-657.

Pertaub DP, Slater M, Barker C (2001) An experiment on fear of public speaking in virtual reality. Stud Health Technol Inform 81:372-378.

Pertaub DP, Slater M, Barker C (2002) An experiment on public speaking anxiety in response to three different types of virtual audience. Presence-Teleoperators and Virtual Environments 11:68-78.

Ramachandran VS, Gregory RL (1991) Perceptual filling in of artificially induced scotomas in human vision. Nature 350:699-702.

Sacks O (1998) The Man Who Mistook His Wife for a Hat: And Other Clinical Tales: Touchstone.

Sanchez-Vives MV, Slater M (2005) From Presence to Consciousness Through Virtual Reality. Nature Reviews Neuroscience 6:8-16.

Schroeder R, Steed A, Axelsson AS, Heldal I, Abelin A, Widestrom J, Nilsson A, Slater M (2001) Collaborating in networked immersive spaces: as good as being there together? Computers & Graphics-Uk 25:781-788.

Slater M, Wilbur S (1997) A framework for immersive virtual environments (FIVE): Speculations on the role of presence in virtual environments. Presence-Teleoperators and Virtual Environments 6:603-616.

Slater M, Sadagic A, Usoh M, Schroeder R (2000) Small-group behavior in a virtual and real environment: A comparative study. Presence-Teleoperators and Virtual Environments 9:37-51.
Slater M, Pertaub D-P, Barker C, Clark D (2004) An Experimental Study on Fear of Public Speaking in a Virtual Environment. In: IWVR 2004. Lausanne, Switzerland.
Steed A, Spante M, Heldal I, Axelsson A-S, Schroeder R (2003) Strangers and friends in caves: an exploratory study of collaboration in networked IPT systems for extended periods of time. In: Proceedings of the 2003 symposium on Interactive 3D graphics, pp 51-54. Monterey, California: ACM Press.
Turati C, Simion F, Milani I, Umilta C (2002) Newborns' preference for faces: what is crucial? Dev Psychol 38:875-882.

Chapter 7
Dramatic Expression in Opera, and Its Implications for Conversational Agents

***

W. Lewis Johnson
Director, Center for Advanced Research in Technology for Education (CARTE)
USC / Information Sciences Institute
4676 Admiralty Way, Marina del Rey, CA 90292 USA
Johnson@isi.edu

1. Introduction

It is commonly agreed among embodied conversational agent (ECA) researchers that ECA behavior should be based upon principles of human face-to-face communication (Cassell et al., 2000; Traum & Rickel, 2002). It is less commonly acknowledged that principles of human *acting* can inform the design of ECA behavior, particularly in making behavior engaging and understandable. Character animators, in contrast, understand clearly the relationship between character behavior and acting (Porter, 1997), and have articulated principles such as exaggeration and staging that are based in part on observations of actors (Thomas & Johnston, 1981; Lasseter, 1987; Maestri, 1999). However, we cannot expect to capture principles of dramatic portrayal in ECAs simply by copying the techniques of animators. ECAs are being developed for a applications with a variety of media characteristics; we therefore need to draw lessons from a range of dramatic media, including those involving live action. Some ECA developers try to incorporate dramatic aspects by collecting motion capture data from actors (Churchill et al., 2000). This approach relies upon the actor's expressive skills to achieve the desired dramatic effect. Unfortunately there is no assurance that motion capture data will appear equally expressive and appropriate when transferred to different media and different dramatic contexts.

This article considers dramatic portrayal from a personal perspective: that of a practicing opera singer. Through examination of the process of preparing and performing an operatic role, I will attempt to draw lessons that may be of value to the design of conversational agents, and discuss how those lessons apply to specific examples of conversational agents. Lessons learned here are particularly applicable to ECA applications dealing with emotional or stressful subjects, those that involve long-term interaction with agents, and those that seek to engage the user deeply in the subject matter. To those of you who are not well acquainted with opera as a dramatic form and doubt its relevance to ECAs, I suggest that you try to suspend your disbelief, and read on.

2. Case Study: *Susannah*

I recently completed a stint as an opera singer, performing the role of Olin Blitch in Carlisle Floyd's opera *Susannah*. The production was mounted by Ventura College, and performed by a mix of professionals and amateurs from the Ventura, Santa Barbara, and Los Angeles areas. The story of *Susannah*, based upon the biblical story of Susannah and the Elders, is set in the Bible Belt of Tennessee. The elders of the local church chance upon Susannah bathing naked in a nearby creek, and accuse her of being a sinful woman. They inform an itinerant preacher, Olin Blitch, who has just come into town to lead a revival meeting, and Blitch resolves to try to convince Susannah to repent. Susannah refuses, because she believes that she has done nothing wrong. Blitch takes her refusal as evidence that she is beyond redemption, and proceeds to seduce her (Figure 1). Only then, when he discovers that she is a virgin, does he realize that Susannah was unjustly accused, and that he is the one who now faces eternal damnation. Floyd termed his work a "musical drama," and it is clear that in this work dramatic goals are paramount. The use of multiple modalities, including music, enhances the dramatic portrayal. On the other hand, the performer's need to convey intent at a distance from the stage, and the need to be heard over a powerful orchestra, make it necessary to intensify expression at times. But because the dramatic expression is intense, the mechanisms involved in its portrayal are relatively easy to discern and study.

Figure 1. Blitch and Susannah

3. Points of Comparison: Carmen's Bright IDEAS and MRE

This article will make frequent references to two particular ECA systems that I have been involved in developing: *Carmen's Bright IDEAS* (CBI) and Mission Rehearsal Exercise (MRE). I will describe these systems briefly here; for more detail please see the cited references.

Carmen's Bright IDEAS is an interactive pedagogical drama designed to teach mothers of pediatric cancer patients to cope better with their problems (Marsella et al., 2003a; Marsella et al., 2003b). It dramatizes the problems of Carmen, a fictional mother of a child with cancer, and shows Carmen discussing her problems with a counselor, Gina. Agent technology is used to determine Carmen's and Gina's actions, and neither character's behavior is scripted ahead of time. *CBI* was developed with dramatic concepts in mind; the story was developed initially as a linear script by a professional scriptwriter, and then extended into a library of possible actions for each character. Character gestures and facial expressions were designed to ensure that the intent of the characters is clear to the viewer (Figure 2).

Figure 2. Points of comparison: *Carmen's Bright IDEAS* and Mission Rehearsal Exercise

Mission Rehearsal Exercise (MRE) is designed to train military leadership skills in stressful situations (Swartout et al., 2001). MRE was developed by USC's Institute for Creative Technologies, in collaboration with CARTE and USC's Integrated Media Systems Center. MRE places trainees in a simulated peacekeeping situation where they must interact with simulated platoon members and make decisions. The action takes place on a large floor-to-ceiling panoramic display that gives an illusion of presence in the virtual scene. The scenario consists of three main scenes. The first scene gives a first-person view of driving into town. The second scene takes place in the town where a traffic accident has occurred between a military vehicle and a civilian car (Figure 2). The climax of the action takes place here. Finally there is a third scene consisting of a fictional television news report summarizing the outcome of the scenario.

4. Some Observations and Lessons

4.1 Dramatic Structure

Large-scale works such as operas have a dramatic structure that helps to promote audience engagement. Individual scenes such as the revival meeting in *Susannah* have a progressive build-up of dramatic intensity. Likewise the sequence of scene leads to the overall climax of the work. Freytag suggested a canonical form for the dramatic structure of such works, called

"Freytag's triangle", which consists of rising action leading to a climax, followed by falling action leading toward the conclusion (Freytag, 1898, cited by Laurel, 1991). Yet simple structures such as Freytag's triangle do not however capture the full structural complexity of operatic works. Instead, action develops over a series of intermediate climaxes, often followed by contemplative scenes in which the characters reflect on what just happened and decide what to do next.

Dramatic structure was also a factor in the design of CBI and MRE. They employ a three-scene structure, in which the main scene provides the climax. A major challenge in these systems is ensuring that each session exhibits proper dramatic structure regardless of what actions the autonomous characters and the user take. For example, Gina supports the dramatic structure of CBI by guiding Carmen through the problem solving steps, when she feels that Carmen is ready to continue. If Carmen, under the influence of the human learner, veers away from the intended dramatic structure, e.g., by losing confidence and refusing to develop options, Gina tries to motivate Carmen to get back on track.

One thing that is needed in titles such as CBI and MRE is a dramatic structure that links multiple sessions. We want each session to have a dramatic resolution, and yet motivate the learner to continue to work through multiple training sessions. This is a common challenge for agent-based applications that interact with users over multiple sessions. Serialized dramatic forms might serve as useful models here. But it may also be possible to adapt the operatic technique of interspersing intermediate climaxes with more reflective scenes, to examine what has just happened and prepare the user to continue the story in subsequent sessions.

4.2 Character Development

In addition to the overall dramatic arc of the story, operas typically also incorporate development arcs for each main character. In *Susannah* each character arc starts with a clear expository scene, in which the character expresses thoughts and intentions, so that the audience understands the motivations for his or her subsequent actions. As the story unfolds more facets of the characters' personalities may be revealed, as they react to new events. We see Blitch go through a series of changes, from upright preacher to seducer to repentant sinner. Floyd has written arias for Blitch at each major change, in order to make the changes clearer to the audience. This is important because, as dramatic theorists since Aristotle have noted, the character's actions should follow causally from the character's traits, and that the character's traits should be consistent throughout (Telford, 1961). So if a character changes over time there must be a cause for this change, from the audience's perspective; these changes may be a consequence of significant plot events, or may reflect additional character traits that have not yet been revealed. When the character's traits are pulling him in conflicting directions, e.g., when Blitch decides whether or not to seduce Susannah, the audience must see that conflict, so that characters actions do not appear arbitrary.

To build character arcs into conversational agents, we need to keep the agent's character traits consistent, while providing mechanisms for development and change. There have been significant advances recently in defining agent character traits; Rist et al. (2003) have developed a toolkit for specifying personality traits in accordance with Dignum's Big 5 model. Gratch's Émile system models emotions using a plan-based model of emotional appraisal, and shows how small biases in emotional appraisal and response can lead to large systematic differences in character behavior (Marsella et al., 2003a). These mechanisms have been integrated into MRE,

yielding agents that can respond in different ways to external events, based upon personality parameters (Marsella & Gratch, 2003). Nevertheless, these models draw a strict dichotomy between agent moods, which are ephemeral, and personality traits, which are fixed. The middle ground, of evolvable yet consistent character traits, needs further development in ECAs.

One potentially valuable mechanism comes from research on the psychology of motivation, which has identified a number of motivational factors that contribute to learning and achievement, such as confidence (Lepper & Henderlong, 2000) and fear of making mistakes (Linnenbrink & Pintrich, 2000). These factors are persistent, but are influenced by events.

Another possible technique is to model explicitly the "front" that characters present to others. Goffman (1959) has observed that people in social situations try to present themselves in a manner that is appropriate to that situation. They attempt to manage both the expressions that they *give*, i.e., the communicative acts aimed at specific people, and the expressions that they *give off,* i.e., actions that others treat as symptomatic of them, that help others to form an impression. A character like Blitch is very much involved in presenting fronts to people, for example when he arrives in town and tries to assume spiritual leadership of the town. When I played Blitch in this context I made a point of conveying to the townspeople a confident, charismatic, empathetic, commanding persona, through his posture, his hand and facial gestures, and his stance and interpersonal distance during conversation. Later in the drama I had Blitch drop that persona, both in his interactions with Susannah and in his confession to God. Finally at the end Blitch tries to reassume his preacher persona in dealing with the townspeople, but he can no longer do it convincingly because he knows in his heart that it is false. Thus a character arc develops through a progression of social stances combined with changes in beliefs and attitudes. In the process, the audience gets to see to some extent behind the fronts that the character presents, and develops a consistent understanding of the character.

4.3 Verbal Expression

Although many of the details of operatic portrayal are written into the score, many other details are omitted, and are up to the conductor, the director, and the singer-actors to create. I will focus here on the verbal aspects of operatic portrayal (i.e., song and speech), emphasizing those aspects that are relevant to conversational agents; nonverbal aspects will be discussed in the next section. *Susannah* employs a wide range of verbal delivery, including conversational speech, half-sung *Sprachstimme,* sung recitatives imitating conversational cadences, and arias.

An important aspect of any verbal expression in opera is its emotional content. Singer-actors have multiple means at their disposal for expressing emotion, including tempo, volume, pitch range, accents, phrase shape, vocal color, and even vocal gestures such as sighs and tremors. The dynamic markings in the score only provide a rough guide to these qualities, and omit important details. But even they indicate characteristics such as volume they do not indicate the underlying rationale for the dynamics. In order for the dynamics to be convincing a performer should infer or imagine the intent underlying the dynamic marking, and try to express the intent.

This expression of intent is not simply a matter of displaying emotion—this emotion must be *communicated to* somebody. Emotional displays arise in the process of communicating to other characters. The manner and intensity of the emotional displays depend upon whether the singer-actor is communicating to an individual or a group, and the degree of familiarity of the listeners. Emotions can sometimes be displayed deliberately, to make the communication more persuasive. The context and communicative goals of expression are important because they influence both the focus of emphasis and the intensity of delivery. For example, when Blitch says to the church elders "Make restitution *now,* brethren!" he is not simply expressing anger, but is angrily uttering a command to them. This causes the entire utterance to be delivered at high intensity, with particular emphasis on the word "now." Intensity is also sensitive to the dramatic structure of the scene; if dialog is leading up to a climax, expressive intensity may increase accordingly.

Dramatic verbal expression poses serious challenges for conversational agents. Speech synthesis techniques that offer expressive variability usually have low speech quality. The text-to-speech synthesizer developed for MRE, in contrast, has good expressive qualities and overall sound quality, while providing significant expressive variability. It is a concatenative unit selection synthesizer that combines multiple limited-domain synthesizers, each specialized to a particular class of communicative intent (informing vs. inquiring vs. commanding) (Johnson et al., 2002). This helps to ensure that each utterance conveys the most suitable basic category of intent. We have recently extended the synthesizer to generate appropriate boundary tones depending upon the dialog context, and to emphasize particular words.

4.4 Dramatic Gesture

Operatic portrayal employs a variety of nonverbal gestures—hand gestures, facial expressions, head and body poses, and body movement. Gestures complement the voice, making intent clearer and more compelling, and they extend portrayal through silent periods, when other singers are singing, or during musical interludes.

Gestural portrayal must work within strict constraints. Temporal constraints are imposed by the musical score, as interpreted by the conductor. Spatial constraints come from the blocking of the scene, requiring action to take place at set points on the stage and movement to proceed from one point to another. The singer-actors must determine what actions to perform and how within these constraints. I will not discuss the issue of blocking design here, but note that it is a complex problem, both for operas and for ECAs, particularly in multi-character scenes.

A major question is what range of gestures to use—should they be based on natural expression or stylized in some fashion? Contemporary singer-actors usually base their gestures on natural face-to-face conversational gestures. The main difference from normal conversation is that actors make greater use of their full body in conveying emotions and attitudes. I used posture extensively in my portrayal of Blitch, to depict his progression from confident man of God (erect, chest thrust forward) to repentant sinner (stooping, slope-shouldered).

Gestures must be natural, fit the constraints of score and blocking, convey intent effectively to the audience, as well as be aesthetically pleasing. Some experimentation may be required during practice and rehearsal to come up with a series of gestures that works most effectively. The danger, as Stanislavski (1936) has noted, is that the gestures take the place of the intention that the gestures are meant to express; the actor "represents" the part, instead of "living" it. What is

required, according to Stanislavski, is an integration of inner intention and outer expression. I submit that this integration is important for conversational agents as well. One advantage of this approach, as Stanislavski has noted, is that it facilitates improvisation. If an actor memorizes a particular sequence of gestures to perform, that makes it difficult to adapt the portrayal if the drama unfolds in a different way from what was anticipated. Unexpected events can happen on stage, even in performances of linear dramas. The unexpected is even more likely to occur in nonlinear, interactive experiences. Similarly if an ECA is simply playing prerecorded gestures, acquired through motion capture or other means, without a model of the underlying emotional state, then if the situation changes unexpectedly the gestures may no longer appear appropriate.

4.5 Give and Take

When multiple players are on stage, as is usually the case in opera, special considerations arise. There are often multiple activities going on at once, which is confusing since a viewer can only focus on one activity at a time. It is important for actors to coordinate their activities to make the overall action on stage understandable and coherent.

One way to lend coherence to multi-player action is to *give focus.* If one player has the primary role in the current action, then the other characters should direct their attention to that character. This helps the audience to see where to direct their attention, and avoids extraneous action on stage that can distract the viewer. We utilized this technique in the opening scene of *Susannah*. As Blitch, I made my entrance quietly, sat down, and listened to one of the townspeople, Mrs. McLean, talk about Susannah, whom she believes is evil. At this point I was giving focus to Mrs. McLean. Then after this Mr. McLean saw me, noted that I was a stranger, and asked me what my name was. I then stood up and announced in my first aria, "I am the Reverend Olin Blitch…" At this point everyone on stage directed their focus toward me, some turning to look at me and listen, some moving downstage so that they can get a better view of me.

Giving focus does not simply involve staring at other cast members, however. Each player must have an intention at all times, and display to that intention. So if a player is focusing on another player and listening to what that player is saying, the first player should react to what the other player is saying, and display that reaction. Action on stage involves a continual give and take among the players, where action leads to reaction which entrains further action. When done right these actions and reactions combine into a continuous flow, which propels the drama forward.

In order for give and take to work most effectively, the two players must work together so that each action provides preparation for the reaction. One mechanism of achieving this is through eye contact. If one player speaks or sings a line that calls for a strong reaction from the other player, he or she often will establish strong eye contact with the other player. This helps the other player to prepare to react to the action, and helps make the focus of action clear to the audience.

CBI and MRE both illustrate how give and take could apply to embodied conversational agents. During the vehicle accident scene in MRE a number of characters are present, but it is hard to tell what the focus of the action is. In Figure 2, for example, the mother and the combat lifesaver are focused on the boy, and the sergeant is focused on the viewer. This may be appropriate when the trainee first comes to the scene, but as the sergeant and the trainee plan how to evacuate the child the focus should shift to the trainee and the sergeant. Part of what makes the situation difficult in MRE is that the injured child and the lieutenant are competing foci of attention. In order to avoid an appearance of lack of focus, transitions in focus from one point to another is necessary over the course of the action, in reflection of changes in saliency over time.

5. Taking the Audience's Perspective into Account

Finally, I will discuss some of the ways in which stage action in opera takes the audience's perspective into account. Theatrical performance offers little in the way of direct interaction between the players and the audience. In fact, such interaction is discouraged, because it tends to lead to bad acting, and because it is difficult to establish give and take with an audience. Nevertheless, stage action on stage is carried out so as to make it understandable to the audience, and many of the techniques described above facilitate this. The following are additional ways in which operatic performance takes the audience's perspective into account; these may have relevance to ECAs, particularly those where the point of view of the audience is fixed, or under the control of the user instead of the agents.

One basic requirement is that the action be visible to the audience. Players must work to keep their action visible, particularly in dialog with other characters. An exchange between two characters, Blitch, and Elder McLean, illustrates this. Blitch, standing upstage from Elder McLean, wants to ask McLean a question. McLean will not be able to answer from this position, since it would involve singing upstage. Therefore Blitch needs to combine asking the question with walking downstage, to a position level with or downstage from McLean, and time his asking of the question so that he does not end up singing upstage either.

One way to make action more visible is to adjust body orientation toward the audience. If two players are standing side by side and engaged in conversation, they are each likely to turn slightly outward, rather face each other straight on, and reserve straight-on orientation for points where particular emphasis is required. This works in part because proscenium provides a two-dimensional frame for the action, making distortions of orientation less noticeable. ECA's could use this technique, since computer displays also frame action, and are usually two-dimensional.

Players must also take into account the distance of the audience. Gestures that read well close up may not be noticeable to audience members sitting at a distance. This means that gestures tend to be more pronounced on stage than in face-to-face conversation. ECAs are rarely life size, and in the future may increasingly appear on handheld devices. The problem of making gestures read on a small screen is similar to the problem of making gestures read at a distance.

6. Conclusions

This article has discussed principles, techniques, and methods of dramatic portrayal in opera, and their application to the development of embodied conversational agents. Investigations such as this complement studies of natural human behavior, and offer insights as to how to make such behavior understandable and interesting when adapted for use by embodied conversational agents. However, one should use caution in applying such lessons. The unique characteristics of computer-based media are still being identified and explored. In any case, one must always be careful about applying principles blindly to any artistic form. Such principles are post-hoc analysis of the intuitive skill of great artists; this was as true in Aristotle's day as it is today. We should not let structural principles stand in the way of injecting creativity into the design of ECAs. Opera at its best possesses an element of magic that is difficult to describe, much less analytically reconstruct. We can only hope to achieve a similar result with conversational agents.

Acknowledgments

The author wishes to thank Kate LaBore, Stacy Marsella, and L.T. Pirolli for their helpful comments, and funding from the Army Research Office and the National Cancer Institute.

References

Bickmore, T. (2003). *Relational agents: Effecting change through human-computer relationships.* Ph.D. thesis, MIT.

Cassell, J., Bickmore, T., Campbell, L., Vilhjálmsson, H., & Yan, H. (2000). Human conversation as a system framework: Designing embodied conversational agents. In J. Cassell, J. Sullivan, S. Prevost, & E. Churchill, (Eds.*), Embodied Conversational Agents,* 29-63. Cambridge, MA: MIT Press.

Chekhov, M. (1953). *To the actor: On the technique of acting.* New York: Harper and Row.

Churchill, E.F., Cook, L., Hodgson, P., Prevost, S., & Sullivan, J.W. (2000). "May I help you?": Designing embodied conversational agent allies. In J. Cassell, J. Sullivan, S. Prevost, & E. Churchill, (Eds.), *Embodied Conversational Agents,* 29-63. Cambridge, MA: MIT Press.

Goffman, E. (1959). *The presentation of self in everyday life.* New York: Anchor Books.

Johnson, W.L, Narayanan, S., Whitney, R., Das, R., Bulut, M., & LaBore, C. (2002). Limited domain synthesis of expressive military speech for animated characters. IEEE TTS Workshop.

Lasseter, J. (1987). Principles of traditional animation applied to computer animation. *Computer Graphics* 21(4), July 1987, 33-44. New York: ACM Press

Laurel, B. (1991). Computers as theater. Reading, MA: Addison-Wesley.

Lazarus, R.S. (1991). *Emotion and adaptation.* New York: Oxford University Press.

Lepper, M.R. & Henderlong, J. (2000). Turning "play" into "work" and "work" into "play": 25 years of research in intrinsic versus extrinsic motivation. In C. Sansone & J.M. Harackiewicz (Eds.), *Intrinsic and extrinsic motivation: The search for optimal motivation and performance,* 257-307. San Diego: Academic Press.

Linnenbrink, E.A. & Pintrich, P.R. (2000). Multiple pathways to learning and achievement: The role of goal orientation in fostering adaptive motivation, affect, and cognition. In In C. Sansone & J.M. Harackiewicz (Eds.), Intrinsic and extrinsic motivation: The search for optimal motivation and performance, 195-227. San Diego: Academic Press.

Maestri, G. (1999). Character Animation 2, Volume 1: Essential Techniques. Indianapolis, IN : New Riders Publishing.

Marsella, S. & Gratch, J. (2003). Modeling coping behavior in virtual humans: Don't worry, be happy. In Proceedings of the Second International Joint Conference on Autonomous Agents and Multi-Agent Systems. New York: ACM Press.

Marsella, S., Gratch, J., & Rickel, J. (2003a). Expressive behaviors in virtual worlds. In press.

Marsella, S., Johnson, W.L., & LaBore, C. (2003b). An interactive pedagogical drama for health interventions. *Proceedings of AIEd 2003,* in press. Amsterdam: IOS Press.

Porter, T. (1997). Creating lifelike characters in *Toy Story. SIGART Bulletin* 8, 10-14.

Swartout, W., Hill, R., Gratch, J., Johnson, W.L., et al. (2001). Toward the Holodeck: Integrating graphics, sound, character, and story. In J. Müller, E. André, S. Sen, & C. Frasson (Eds.), *Proc. of the Fifth Intl. Conf. on Autonomous Agents,* 409-416. New York: ACM Press.

Rist, T., André, E., & Baldes, S. (2003). A flexible platform for building applications with life-like characters. In W.L. Johnson, E. André, & J. Domingue (Eds.), *IUI 03: 2003 Int'l Conf. on Intelligent User Interfaces,* 158-165. New York: ACM Press.

Stanislavski, C. (1936*). An actor prepares.* New York: Theater Arts Books.

Telford, K.A. (1961). *Aristotle's Poetics: Translation and analysis.* South Bend, IN: Gateway Editions.

Thomas, F., and Johnston, O. (1981). *The illusion of life: Disney animation.* New York: Walt Disney Productions.

Traum, D. & Rickel, J. (2002). Embodied agents for multi-party dialogue in immersive virtual worlds. In C. Castelfranchi & W.L. Johnson (Eds.). *Proc. of AAMAS,* 766-773. New York: ACM Press.

Chapter 8
Human Activity Behavior and Gesture Generation in Virtual Worlds for Long-Duration Space Missions

Maarten Sierhuis, William J. Clancey,[2] Bruce Damer,[3] Boris Brodsky, and Ron van Hoof[5]

***

[1] RIACS/NASA Ames Research Center
[2] NASA Ames Research Center/IHMC
[4] SAIC/NASA Ames Research Center
[5] QSS/NASA Ames Research Center
Moffett Field, CA 94035

[3] DigitalSpace Corporation.
Santa Cruz, CA

Abstract

A virtual worlds presentation technique with embodied, intelligent agents is being developed as an instructional medium suitable to present *in situ* training on long term space flight. The system combines a behavioral element based on finite state automata, a behavior based reactive architecture also described as subsumption architecture, and a belief-desire-intention agent structure. These three features are being integrated to describe a Brahms virtual environment model of extravehicular crew activity which could become a basis for procedure training during extended space flight.

1. Introduction

Future space flight will increasingly be longer and more complex. When space missions become longer—years as opposed to months—and include humans, the training of human astronauts will be increasingly more difficult. This is not in the least because training has to be done during the mission. Although we will always train our astronauts on Earth in the basics of human space flight, mission specific training and even basic astronaut capabilities on an extended duration mission, such as donning a space suit for an extra-vehicular activity (EVA) on Mars, need also be done during the mission. Even if trained on Earth, after months of space travel previously trained tasks will be forgotten or at minimum need to be reviewed. Mission critical training will have to be done as just-in-time training during the mission. Training is one of the critical elements in safety for human space flight. Today NASA trains its astronauts for years in close to real-life training simulations of Space Shuttle and International Space Station missions. Full-fledged vehicle and mission control simulations are done for months, if not years before every mission. While training is an important aspect of mission preparedness, today's ISS astronauts have very little training in some of the most basic procedures, such as medical emergency and maintenance procedures. When new science experiments are delivered to the ISS, astronauts often have not been trained to perform these experiments. Therefore, on the job just-in-time training is already a fact of astronaut life today. As NASA missions will continue to be longer, this issue is more likely to become more difficult. We are convinced that if the training issue for

long-duration space flight is not solved, humans will always be limited in how long they can be in space.

One way to address the mission-training problem is to use immersive training facilities onboard the spacecraft and provide the astronaut with just-in-time training during the mission. We believe that virtual inhabited 3D-spaces provide a potential training solution for long-duration space flight. Virtual inhabited 3D-spaces or virtual worlds provide a potential solution to the issues of just-in-time training during long space missions, because:

a) Virtual world systems can be relatively small[2], which means that they can be provided in the small spacecrafts where space will always be a constraint.
b) Virtual world are collaborative VR environments where people can enter and interact with the virtual world and each other.
c) Virtual world systems can provide situation-specific training solutions, because real-life teamwork situations can be simulated inside the virtual world system.
d) Virtual world systems consist largely of software, which means that the same system can be a platform for any training domain that is needed for the mission. This allows the mission developers to develop training modules for almost any mission task.

Virtual worlds (VW) are three-dimensional virtual spaces inside a computer connected to the internet that present as "stages" with objects and characters (avatars and autonomous agents or softbots). The user enters the virtual world via the Internet as an avatar[3], a representation of the human user through which the user experiences and interacts with the virtual world, its bots and other avatars. A distinct difference between virtual worlds and immersive virtual reality (VR) systems is the way the user enters the virtual space. Most VR systems are standalone environments not shared via the Internet. In immersive VR systems the user enters as him or herself. There is no representation of the user inside the system. It is as if the user is inside the system, but is not a participant in the system. Computer games use an intermediate form of user interaction. In computer games the user enters the system as him or herself, but have a limited set of interactions they can choose from; Shooting at the enemy, driving a car, flying an airplane, et cetera. Some games, most notably sports games, allow the user to interact with the game world by controlling one or more team player bots in the game. In a VW the user enters as his or her avatar, i.e. a virtual representation of him or herself. It is as if the user could be a player on the team in the game. The VW is a shared virtual environment on the Internet. The user becomes a participant in the virtual world in that he or she cannot only interact with the other characters in the VW, but the other characters (e.g. other users entering the world via the Internet) can "see" and interact with the user by interacting with their avatar. Another difference between immersive virtual reality and virtual worlds is that the objective of a VW is not to make the user forget about virtuality and make them believe that they are in the real world. Instead, the objective is to create parallel worlds for the user. One of the reasons virtual worlds are popular is that they exist on the internet and allow ordinary people to come together in a virtual meeting place with a basic computer setup (an Internet browser or a VW client program) and an Internet connection. VW

[2] *They run on laptops in internet browsers, and do not need large VR environments.*
[3] *The original meaning of the word avatar stems from the Sanskrit word avataraa which means "descent." The word comes from a Hindu myth about the incarnation of the deity Vishnu. A more contemporary Western meaning of the word is "a temporary manifestation or aspect of a continuing entity."*

systems are not the same as other virtual reality systems in that people are entering virtual realities that do not exist outside the computer to meet and interact with other people (Damer, 1997). One interesting consequence and opportunity of this difference is that it becomes possible to think of the virtual world as an extension of the physical world of the user. Eventually this could allow a seamless integration of the real and the virtual world for a realistic team-training scenario.

1.1. Objectives

The research described in this paper has as its ultimate objective to create a development environment for creating uploadable just-in-time VR training applications for crews on their way to or on the moon or Mars. With the term uploadable we mean training applications that can be created on Earth and uploaded via space communication networks to the crew for just-in-time delivery. This would allow training modules to be developed during the mission and delivered in time for training of the crew. This is an important feature, because it gives mission developers the flexibility to continue development of procedures and mission tasks while the mission is in progress. It is advisable that for long-duration missions not every aspect of the mission has to be completely finished before the mission. For example, while the human crew is on its way to Mars specific surface tasks and experiments can still be in development, or being tested on Earth. Also, during a long-term mission situations will develop that need changes in systems and procedures. New training material will then need to be developed and uploaded to the crew to handle these situations.

We are researching the development of a general VW training development environment for the creation of mission critical crew training applications that can be delivered to the crew just in time for the needed training. Our idea is that with this environment it will be possible to develop a wide-range of virtual reality training applications for the crew on long-duration missions. To give an idea of different possible training domains for crews, think about training the crew on putting on (donning) space suits where two or more astronauts are needed to accomplish the task, training the crew on maintaining the green house on Mars or maintain the habitat life support systems. Other examples are the training of team tasks where humans and autonomous robots have to coordinate the work.

1.2. Technical Challenge

Today's advances in virtual worlds allow the developer of the virtual world to create semi-autonomous agents (bots) that have some predefined behavior. The behaviors are mostly created in a scripting language. These bot-scripting languages are low-level languages that do not easily allow for the development of sophisticated bots that have a flexible set of behavioral capabilities enabling them to react to previously undetermined situations. Today, most bots are simple finite state automata (FSA) that have a small set of well-defined behaviors, based on simple perceptual inputs and state transitions (Madsen, Pirjanian, & Granum, 1999).

Agent behavior in a Virtual World

The challenges that we are addressing deal with the aspects of avatars and softbots—or simply bots—as more or less autonomous agents, their perception and behavior in and of the world that they "live". Madsen and Granum (Madsen & Granum, 2001) describe three different ways of developing the behavioral component of autonomous avatars and bots within a VW:

1. A behavioral component implemented by an event-based finite state automaton (FSA). This approach is what is used in most current VW environments (e.g. Active Worlds (ActiveWorlds, 2004), Adobe Atmosphere™ (Atmosphere, 2004), Ogre (Ogre, 2004)), and is supported with a "built-in" scripting language (e.g. C libraries or Java-script). The down side of the FSA approach is the difficulty in developing flexible artificial intelligence (AI) like behavior. Scripting languages, although powerful, are not suited to represent complex behavior, for the same reason that imperative programming languages are not suited for implementing AI systems, and declarative programming languages were developed for this purpose.
2. A more flexible approach is to use a behavior-based reactive architecture, more commonly referred to as a subsumption architecture. Madsen and Granum describe the Blumberg Agent Architecture for the development of their Bouncy virtual world agent (Blumberg, 1996). This architecture is based on Rodney Brooks' subsumption-based robots (Brooks, 1991) (Brooks, 1997).
3. A third approach is using a belief-desire-intention (BDI) agent architecture, such as JAM (Huber, 1999) and dMARS (d'Inverno, Luck, Georgeff, Kinny, & Woodbridge, 2004). BDI-architectures have their roots in distributed artificial intelligence, in which agents operate with explicit plans and goals that are activated based on pre- and post-conditions in production rules matching on the beliefs of an agent. The beliefs of an agent are symbolic representations of the agent's correspondence to information it has about the world.

Madsen and Granum describe these three approaches as possible alternatives for describing avatar behavior. From our experience in modeling human behavior, we have independently come to the conclusion that there is a need for combining all three approaches in one unified avatar and bot behavior model. Figure 4 shows how the behavior of an autonomous avatar or bot can be divided into three behavioral components.

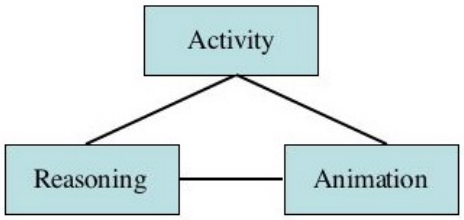


Figure 4. Model of autonomous avatar or bot behavior

The animation component describes the behavior of the agent (an autonomous software agent) within the graphical world. In this component the externally visible behavior of the agent is generated. This component is what distinguishes a bot from a software agent. Simple actions are visualized with scripted animations of a virtual character. These animation scripts are created using a FSA approach, usually using the VW environment's scripting language. Complex behaviors are visualized by combining basic animation scripts into a more complex behavior. For example, to visualize an agent sitting down and drinking a cup of coffee we sequentially combine an animation of a bot sitting down and reaching for an object, with an animation of the bot grabbing the coffee cup object and then bringing the coffee cup to its mouth, and lastly an animation of showing the bot drinking from the cup. The resulting animation is a visualization of the agent's drinking from the cup activity.

Figure 5. Astronaut crew agent enacting a make and drink coffee animation sequence inside the virtual FMARS habitat (ARC)

The activity component describes how particular behaviors are exhibited. This component describes what behavior in the animation component should be executed, for how long and in what context. The best way for describing the agent's behavior in this component is with a behavior-based subsumption approach. Unlike in a plan and goal-based approach used in most BDI-architectures, a bot's behavior is not always goal-driven. Modeling people as bots within a virtual world requires us to model real-world behavior based on interactions with other bots and avatars. There arise many situations in which people's behavior is not goal-oriented or intention driven. Clancey describes how human behavior is often related to motives and cultural norms (William. J. Clancey, 2002). Imagine standing waiting with a group of people to go outside. While you're waiting you take the opportunity and take a picture of the group. Although you could ultimately describe the waiting activity with a plan and goal approach (as you can describe it with an FSA approach), it seems strange and unnatural to describe the taking of the picture while you're waiting as a sub-goal or plan of the "waiting plan." A more flexible and natural approach to describing the "waiting activity" is a hierarchical subsumption-based organization of competing activity behaviors where, for example, the competition between the "standing still and do nothing" and the "take a picture while I am waiting" activities is decided based on a combination of reactive and motive-driven reasoning paradigms.

The reasoning component describes when particular behaviors are exhibited. To continue with the "taking a picture while you are waiting" example, this component determines how the competition between the "do nothing" and "take a picture" activity is resolved in a particular situation. Behavior is situated and it depends on the situation in which the waiting occurs how one would behave. The reasoning behavior of the agent described in this component correlates the agent's individual perception of the situational context with the possible activity behavior, such as where the waiting occurs, what role the agent believes it plays in the group, the purpose of the group going outside, the agent's motive, the group norms and culture, et cetera. Representing this type of reasoning behavior is best done using an agent-based BDI-approach in which an agent has its own belief-set. An agent can reason over its belief-set and the result of this reasoning behavior could be a decision for what activities in the activity component to execute. The reasoning component should be reactive as well as deliberative. A belief-based approach allows both, since the way an agent can obtain new beliefs in its belief-set can include not only forward or backward reasoning, but also the assertion of new beliefs via communication with other agents and perception of real-world events, enabling reactive behavior.

The Brahms Virtual Environment (BrahmsVE) uses this three-component model for representing autonomous avatar and bot behavior. We describe the implementation of this behavior model in more detail in subsequent sections in this chapter.

2. The EVA Prep scenario

This section describes a scenario of a BrahmsVE model, called FMARS, developed in 2002. The FMARS model consisted of two independent scenarios from Crew 1 of the Flashline Mars Arctic Research Station (FMARS) in which a crew of six people lived and worked for a week, conceptually as well as physically simulating a Mars surface mission (W.J. Clancey, 2002) (Clancey, Sierhuis, Damer, & Brodsky, in press). The scenario is about the activities and coordination of three crewmembers during the preparation activity for an Extra-Vehicular Activity (EVA) outside the FMARS habitat. We refer to this as the EVA Prep scenario. A Brahms model was developed for this scenario, showing what the crew did to get ready for an EVA and donning their space suits. The scenario was developed based on video clips and photographs—taken by Clancey, who was one of the FMARS crewmembers—of an actual EVA preparation activity during the Crew 1 mission. Clancey's research as an ethnographer during this crew rotation was to study the daily life of the FMARS crew (Clancey, 2001).

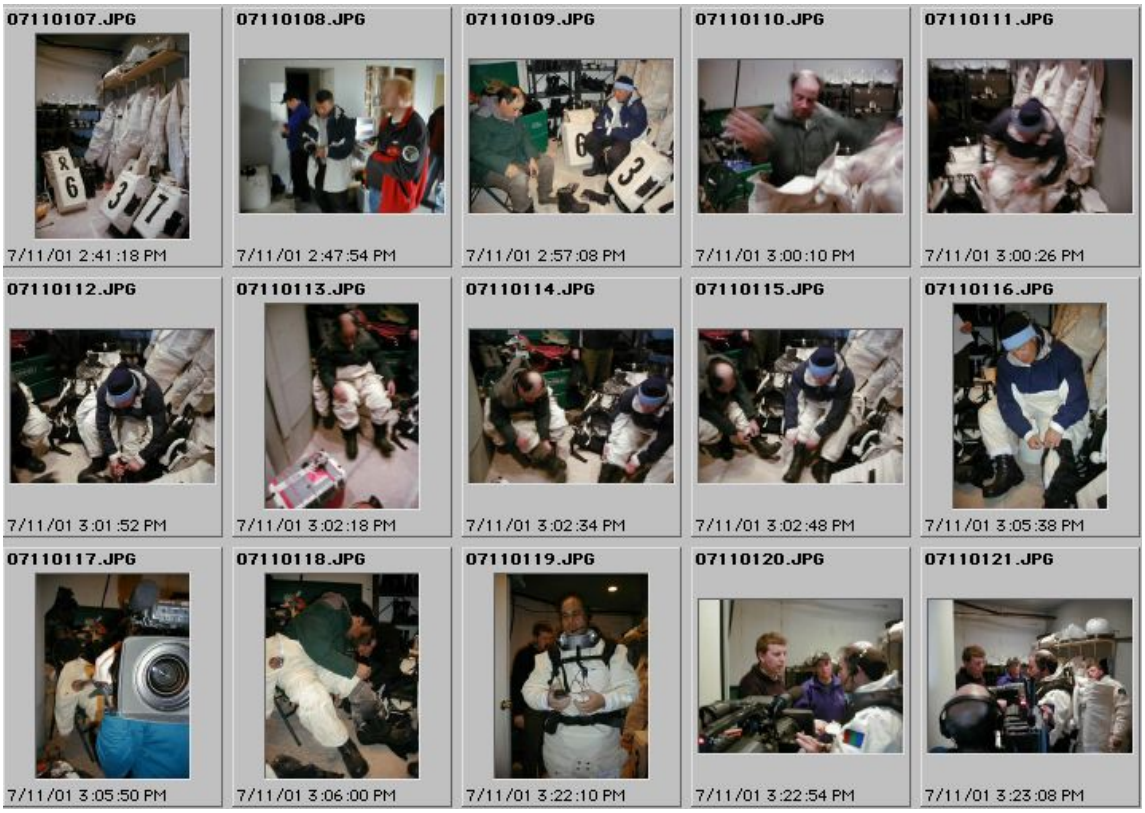


Figure 6. Clancey's chronological EVA Prep photograph record

The resulting EVA Prep model was the culmination of Clancey's ethnographical analysis of the EVA preparation activity, Brodsky's representation of this analysis in a Brahms model and Damer's virtual world design team's creation of the Adobe Atmosphere FMARS habitat Virtual World and the OWorld bot animation models.

One of the first activities in the design of the EVA Prep scenario is the description of the scenario in plain English (the reader should refer back to Figure 6 while reading this description in Table 1):

Table 1. Scenario description in plain English

Notation:

Object – Italicized
Areas – Underlined
Activities – Bolded
Agents – Italicized and bolded

[…]

The **Crew** then proceeds with suit **donning**.

The **Crew walks** to the suit closet area (not really a closet, just hanging there, see pictures), and **finds** her suit.

The **Crew** then **walks** to the shelf area, **selects** boots (not person-specific, selected by size, assume any size is always available).

The **Crew** then **puts on** suit bottom (stepping into suit and pulling it up to pants level. Probably to snapshots would suffice for now).

The **Crew** then **calls** for an available **Helper** for assistance (in putting on boots and gaiters).

The **Helper** responds (assume he is present in the EVA prep room), and **checks** if an empty chair is available. If not, he **brings** a chair (from the wardroom table. Again, don't have to show the lab area and the process of picking. Lets just have the Helper leave the room, and then come back holding the chair). He then **puts** the chair down. (Where exactly to put down the chair is something TBD. Need to settle on a "tiling" solution that wouldn't overburden the simulation with a generic array of areas, but rather just define a few that would be used throughout the simulation, perhaps even reusing the same area for multiple purposes.)

When both the **Helper** and the chair are in place, the **Crew walks** to the chair (located at a pre-specified area, see above), **sits down** on it, and **puts on** boots. (Includes both pulling on and fastening them. For this and subsequent EVA prep fragments, we will need to browse through pictures to find how the Helper actually assists, in this case in putting on buts. That may require some unique gestures. At a minimum, the Crew should be once seated feet on the ground, and another time with one foot in the air, with and without the boot. The Helper would be standing nearby holding the boot, then kneeling holding the boot, then kneeling not holding the boot… Just a speculation).

When the **Crew** has her boots on, the **Helper puts on** gaiters on her.

The **Crew** then **gets up** from the chair, and **puts on** jacket to wear under the suit. (No help necessary).

The **Crew** then **puts on** suit top (pulling the suit up and donning the upper part over torso and arms. (Looks like two pictures: The suit pulled up, and the suit over torso and arms […]).

The **Crew** then **calls** for an available **Helper** for assistance (in zipping up the suit back).

The (available) **Helper walks** over to the **Crew** chair, and **zipps up** the suit back. (Showing the helper standing behind the crew, holding arms up towards the crew's back).

The photographs and the scenario description are important starting data for the development of the Brahms model, as well as the FMARS Virtual World model and the animation models for the bots in the virtual world.

Next, we will describe the three behavioral components of the BrahmsVE architecture, using this scenario as our example.

3. Brahms Virtual Environment

The Brahms Virtual Environment (BrahmsVE) is a combination of different technologies developed at NASA Ames, DigitalSpace Corporation, and Adobe. The environment, shown in

Figure 7, incorporates the Brahms multi-agent language and virtual machine as a bot-language (developed by NASA Ames) for the Atmosphere™ interactive 3D multimedia creation and delivery platform for the Web (developed by Adobe), using the OWorld engine (developed by DigitalSpace).

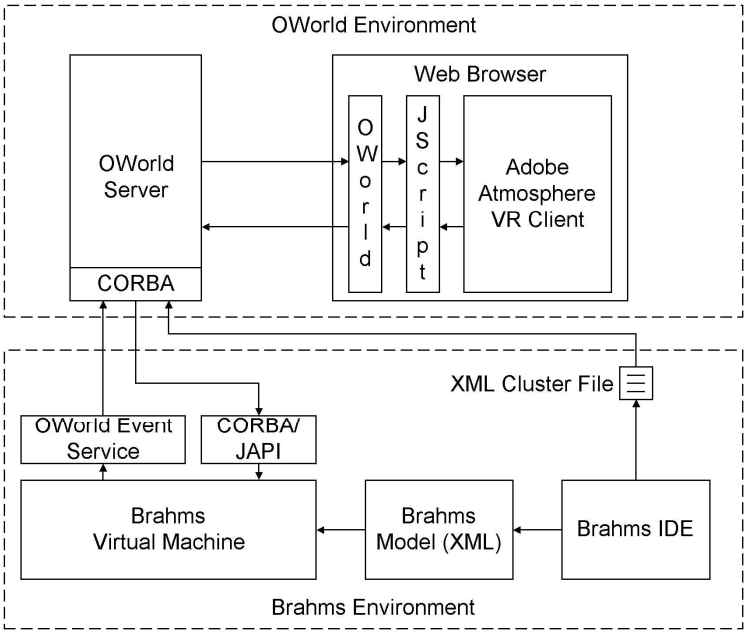


Figure 7. BrahmsVE Architecture

3.1. The Agent Behavior Model—Brahms

Brahms is a multiagent modeling and simulation language, earlier described in (Clancey, Sachs, Sierhuis, & van Hoof, 1998)(Sierhuis, 2001) (William. J. Clancey, 2002) (Sierhuis, Clancey, & Hoof, Submitted). The Brahms environment is in particular developed to model behavioral patterns of people engaged in purposeful activities—what people do—something which has been referred to in the past as people's practice (Schön, 1982)(Lave, 1988)(Suchman, 1987). Without describing the Brahms language in detail—for that we refer you to (Sierhuis, 2001) (Sierhuis et al., Submitted)—we give a description of the important aspects of the Brahms language that allows us to model people's behavior as purposeful and interleaved daily activities. The Brahms language implements the two components of the behavioral model shown in Figure 4—the activity and reasoning components—and is what makes Brahms a good autonomous avatar and bot-language for virtual 3D environment.

Brahms is a multiagent BDI language. Every agent has its own set of beliefs that directs the agent's reasoning and activity components. Agents can communicate their beliefs enabling the reaction of an agent to another agent's communication (in terms of reasoning and/or activity execution). Brahms agents are also reactive to the detection of state changes in and of their world environment. Agents are situated in a modeled geographical world with objects that can be represented with state changing behavior. Object states are modeled as facts in the world, and agents can detect these facts in certain situations, making them beliefs for the agent and allowing them to react to them as soon as they are created. The Brahms engine or virtual machine (VM) is based on an activity-subsumption architecture, driving flexible reactive activity behavior of agents. The workings of this architecture will be discussed next.

The Activity Component in Brahms

Activities are the most important construct in the Brahms language. All agent behavior has to be modeled as an activity. There are three different types: primitive, composite and Java activities. All activities have a user-defined name representing a behavior defined by the modeler. According to our theory of activities (Sierhuis, 2001) (William. J. Clancey, 2002), the name of an activity should be the name of an observed behavior of a person in the real-world that the agent represents, but there is no rule in Brahms that states that the agent has to represent a person and that this has to be a person in the real world. It is the responsibility of the modeler to decide the relevance of the model to the system behavior that is being modeled. This allows the use of the Brahms language in any domain and for any purpose, including, but not restricted to, modeling social phenomena, human behavior, and software agent behavior. Activities have a defined beginning and end determining the duration of the activity. Just as people in the real world, when and how a Brahms agent performs an activity depends on the agent's context in a particular activity, and the actual performance and duration of the activity emerges during the execution of the model, based on the changed belief-state of an agent.

The design of an activity is one of the most important activities of a Brahms modeler. Here we describe the design of the Space Suit Donning composite activity from the EVA Prep model description from Table 1 (see Table 2).

Table 2. Scenario activity design

Notation:

Object and attributes – Italicized
Areas – Underlined
Activities – Bolded
Agents and groups – Italicized and bolded
<Comments> – In angular brackets

<Helpers assist the crew members who are behind (in procedure steps) first>

<Each **EVA_Crew** has an associated conceptual object EVA_Procedure , which attribute values are facts (detected by helpers to know who is behind).>

<Wherever "EVAPrepRoom" is specified as a location, always replace with a more specific one>

<Below, an individual **EVA_Crew** member is denoted **AgentC**, and an individual **EVA_Helper** – **AgentH**. Selected individual objects are, for simplicity, category names suffixed with 'C' (e.g. "HelmetC")>.

SuitDonning <composite activity>

EVA_Crew performs
- **MovingToLocation** (SuitClosetArea) <any better name?>
- **FindingSuit** <SuitC (suits – person-specific)>

- **MovingToLocation** (ShelfArea)
- **SelectingBoots** <BootsC, by size (assume any size is always available)>

- **PuttingOnSuitBottom** <step into suit and pull it up to pants level>

- broadcast **CallingAvailableHelper**
  - <**AgentC**.needHelper = true>
- when **AgentH**.availableToHelp = true
  - **MovingToArea**(ChairCArea)
  - **SittingDown**
  - **PuttingOnBoots** <including fastening boots>
    - < **AgentC** contain BootsC>
  - **HavingGaitersPutOn**

```
                when AgentC contains GaitersC
                        GettingUpFromChair

        PuttingOnJacket <to wear under suit>
                <AgentC contain Jacket>
        PuttingOnSuitTop
                <pulling suit up and donning the upper part over torso and arms>
                <suitTopOn = true>
        HavingSuitBackZippedUp
        when suitBackZipped = true
                FindingTools
                        <RZ has the Shovel,
                        KQ has a Camera around her neck,
                        VP has a Checklist around his neck>
                AttachingToolsToSuit
        PuttingOnCamera <should've been brought in from the stateroom>
        PuttingOnChecklist <both are put around neck>

EVA_Helper performs
        when AgentC.needHelper = true
                when not AgentH.location = EVAPrepRoom
                        MovingToArea(EVAPrepRoom)
                when not (chair.location=EVAPrepRoom) & (chair.available=true)
                BringingChairs <composite activity>
                        MovingToArea(WardroomTableArea)
                        get SelectingChair
                                <AgentH contain ChairC>
                        MovingToArea(EVAPrepRoom)
                        MovingToArea(ChairCArea) <decided on in advance>
                        put PuttingChairDown
                                <AgentH contain ChairC is false>
                communicate ShowingReadinessToHelp
                        <AgentH.availableToHelp = true>

                when AgentC contains BootsC
                        PuttingOnGaiters <over boots, with AgentC still sitting>
                                <AgentC contain GaitersC>

                when AgentC.suitTopOn = true
                        ZippingUpSuitBack
                                < AgentC.suitBackZipped = true>
```

The design of the EVA Prep model is divided in a number of sub-models: Agent-, Object- and Geography Model. The Agent Model defines the agents, the agent groups and the activities. The Object Model defines all the objects in the physical world (e.g. a chair, the spacesuit), as well as the conceptual objects the agents use in the reasoning component (e.g. a procedure). The Geography Model defines conceptual locations that connect to actual locations in the world.

Agents and Groups

The scenario design in Table 2 shows that there are two types of FMARS habitat crewmembers involved in the spacesuit donning activity. The first group includes those habitation crewmembers that are to go on an EVA (the EVAGroup). Obviously, these habitation crewmembers are the ones that will have to don a spacesuit. The second group includes those habitation crewmembers who help the EVA crew to suit up and get ready for their EVA (the EVAHelpers). In Brahms an agent can be a member of multiple groups and inherit from all groups it is a member of. The Agent Model for the scenario in Table 2 is presented in Figure 8.

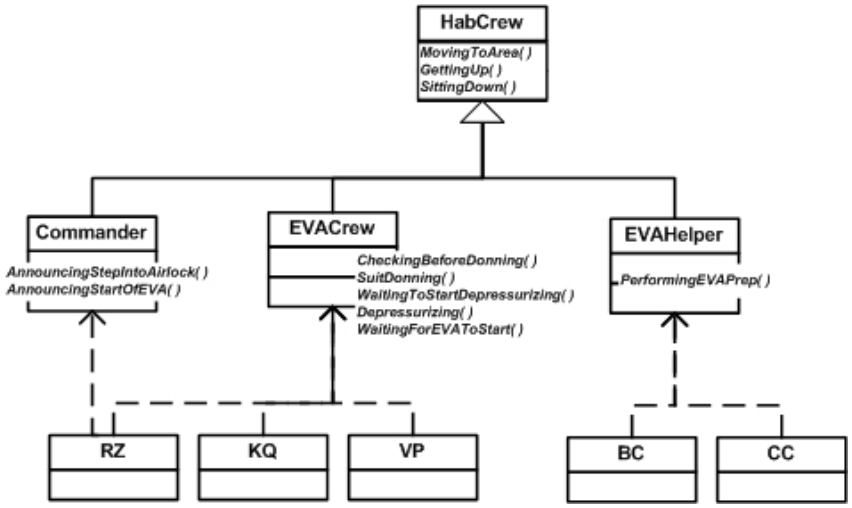


Figure 8. Agent Model

Figure 8 shows the groups and agents in an inheritance hierarchy with the names of the, by the agents inherited group activities. Agent RZ is a member of three groups, HabCrew, Commander and EVACrew, and inherits all the activities from these three groups (the compiler resolves the multiple-inheritance conflicts from the HabCrew group). Agents KQ and VP are members of HabCrew and EVACrew and thus inherit the activities from these two groups, but not those from the Commander group. Agents BC and CC are EVAHelper agents and also HabCrew agents, and thus inherit these group's activities. Table 2 decomposes the SuitDonning( ) composite activity into more detail. Figure 8 only shows the top-level activities, including the SuitDonning( ) activity of the EVACrew group.

Defining the inherited activities in the model does not immediately allow the agents to execute those activities. Defining the activities means that the agent has the potential to perform them, but when they will be performed has to be specified in situated-action rules called workframes. Workframes are also specified in the Agent Model, either at the group or agent level (they are not shown in Figure 8). Workframes are of the form

When (belief-conditions are true)
Do
    Activities, and
    conclude new beliefs for the agent and/or facts in the world
EndDo

Workframes are declarative production rules matching on the belief-set of the agent executing it. This mean that if the belief-conditions in a workframe match a belief in the belief-set of the agent, the condition evaluates to true, and the variables in the condition are bound to the agent or object that is the object of the belief (see (Sierhuis et al., Submitted), (Sierhuis, 2001), and (van Hoof & Sierhuis, 2000) for a detailed explanation of precondition matching). Table 3 shows the PerformSuitDonning workframe in Brahms source code (bold characters are keywords).

Table 3. Workframe PerformSuitDonning

```
workframe PerformSuitDonning {
        variables:
                forone(Commander) commander;
                collectall(EVAHelper) evaHelper;
        detectables:
                detectable NoticeAvailableHelpers {
                        detect((evaHelper.available = true))
                        then continue;
                }
                detectable StartDepressurizing {
                        detect((commander.stepIntoAirlock = true))
                        then abort;
                }
        when(
                knownval(evaHelper.memberEVAHelper = true) and
                knownval(commander.memberCommander = true))
        do {
                CheckingBeforeDonning();
                SuitDonning();
                conclude((current.suittedUp = true));
                AnnouncingSuittedUp();
                WaitingToStartDepressurizing();
        }
}
```

The statements in the body of the workframe (do { } section of the workframe in Table 3) are executed in sequence from left to right and top to bottom. Figure 9 shows the agent execution timeline for agents RZ and CC from one simulation run, displayed in the Timeline View of the Composer application[4]. This timeline shows in what locations the agent is and is moving to at what time—this is shown as the top bar (EVAPrepRoom). You can see that agent RZ moves for a short time to a different location (Shower) during the CheckingBeforeDonning composite activity to fill his water bag.

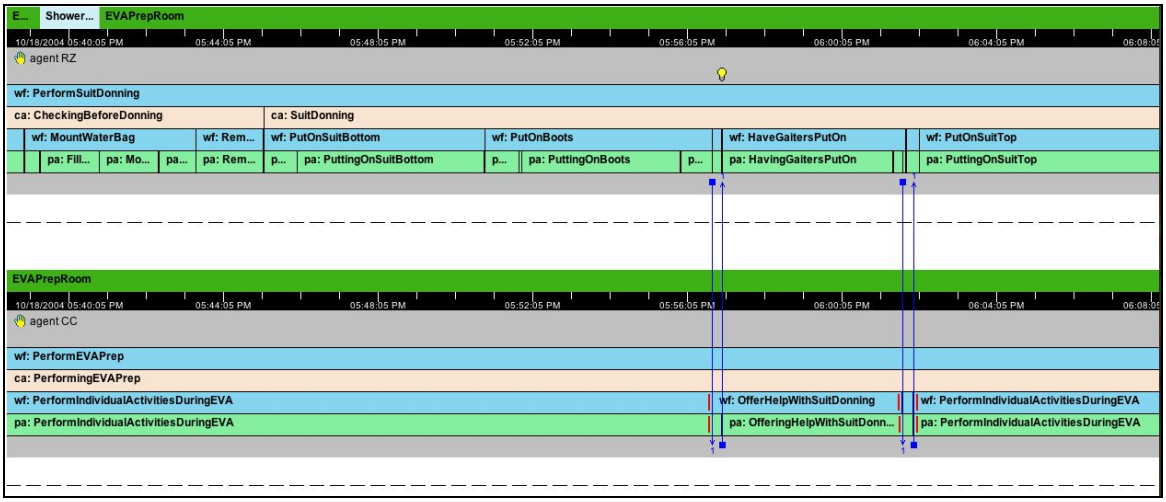


Figure 9. EVACrew and EVAHelper Agent Timeline

[4] *The Composer is an interactive design environment (IDE) for the Brahms language. It is a modeling, programming and simulation execution environment.*

Bellow the location bar you find the time bar. This provides the time at which things occur. The Timeline View allows the user to zoom in and out of the timeline. The time bar in Figure 9 is at a zoom-interval of 2 minutes. This means that every white tick interval on the time bar represents 2 minutes of simulated time for the agent. It is not possible to see in Figure 9, but the total PerformSuitDonning workframe from Table 3 takes 1 hour, 33 minutes and 25 seconds. Bellow the black time bar you find the name of the agent, next to the agent icon that looks like a little hand. Bellow that you see the workframe-activity instantiation represented over time. This provides a hierarchical overview of the workframes and activities that the agent is executing at every simulation clock tick. It shows when and how long workframes and activities within it are being executed. For example, Figure 9 shows that agent RZ is executing the PerformSuitDonning workframe for the entire figure and agent CC is executing the PerformEVAPrep workframe at the same time. These activities are of type composite activity.

Composite activities are activities that are decomposed into lower-level subactivities, workframes and thoughtframes (see Figure 10). A primitive activity is an activity that is not further decomposed. It can be used to represent an action in the world that is not further decomposed. Primitive activities have a specified maximum or random duration. This is different from a composite activity in that it has a pre-specified duration. In contrast, the duration of a composite activity depends on the duration of the subactivities executed within it (note that thoughtframes have no duration—see the small light bulb at the top of the agent RZ timeline in Figure 9).

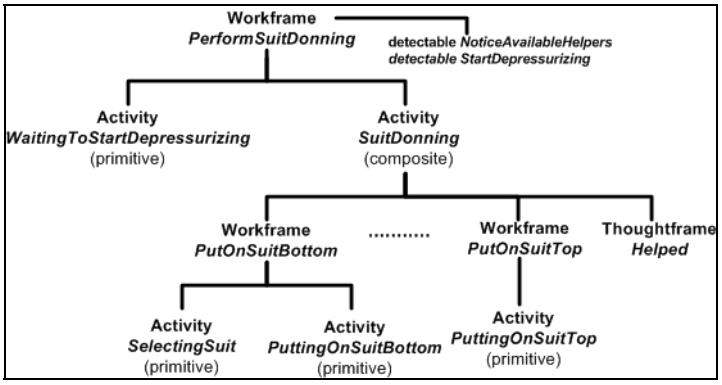


Figure 10. PerformSuitDonning Workframe-Activity Hierarchy

Primitive activity duration is determined at the start of its execution—either randomly chosen or given as a max duration—but is not necessarily the actual duration of the activity. The actual duration of an activity depends on the state of the workframe instance[5] (WFI) in which the activity is being called. Each WFI is in one of the states shown in Figure 11. The state of an agent's activity behavior is defined by the combined sets of available, working, interrupted, and interrupted-with-impasse WFIs at any moment in time.

[5] *When a workframe (or thoughtframe) is fired (i.e. the preconditions are matched against beliefs in the agent's belief-set) a workframe instance is created for every workframe variable context that matches all preconditions. Each workframe instance is now an independent version of the workframe and will be executed independently from each other, with different variable bindings (determined by the WFI-context).*

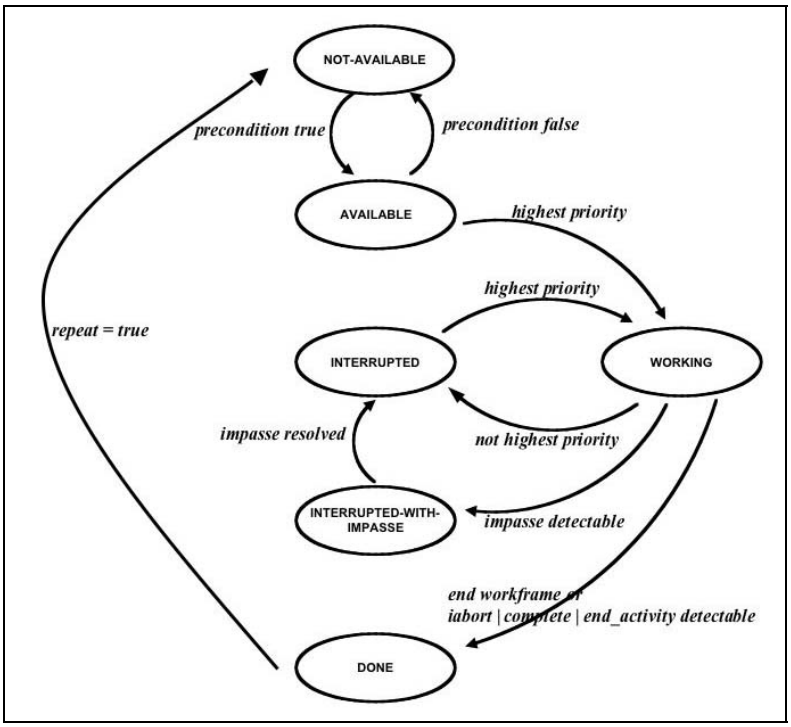


Figure 11. State transition diagram for workframe instances

There can only be one current activity for an agent. The time an activity has been active can only change when the activity is the current activity. Therefore, when an activity is in a non-active state its active time is not increasing, although simulation time is always increasing. Which activity is the current activity depends on which WFI is in the working state and the activity execution sequence of the WFI-body.

There are different ways a WFI can change state. One way is through the use of priorities. Every time a workframe fires the created WFIs receive a priority, based on the priority of the workframe, if given, or the highest priority of the activities called within the workframe body. The default priority is always zero. The work selector process in the agent's inference engine determines which of the available, working and interrupted WFIs have the highest priority (see Figure 12). This one is moved to the working state and is executed by the work executor process. Every time a new WFI becomes available, there exist the potential that the working WFI is interrupted by a higher-priority WFI. In that case the current working instance is moved to the interrupted state, and the new instance with the highest priorities is moved to the working state, and thus becomes the current WFI the agent is executing.

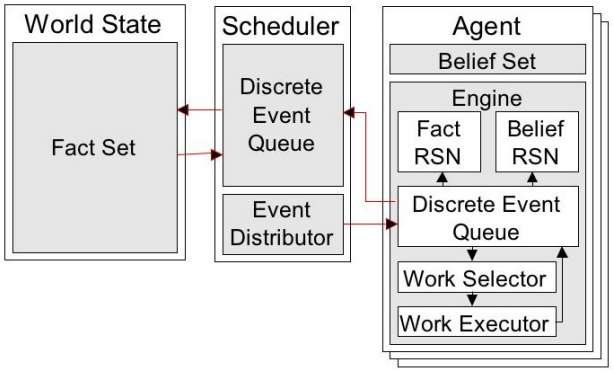


Figure 12. The Agent's Inference Engine

There are other ways for an activity to change from a working state. The state change described above is based on other 'independent' workframes firing. However, a WFI can change its own state. The default way for a WFI to change its working state is when the body is finished executing. At that moment the WFI automatically moves from the working state to the done state and there it gets deleted, or moved to the not-available state if the repeat-clause is set to true. However, there are other state-changing events that can be represented inside a workframe. This is done using a detectable. Table 3 shows the declaration of the NoticeAvailableHelpers and StartDepressurizing detectables. A detectable defines that if the agent detects a fact in the world this fact becomes a belief of the agent (this is accomplished via the event distributor in Figure 12: the fact is part of the fact set in the world state). The belief is then matched to the detect condition in the detectable. If the agent has a belief that matches the condition the body of the detectable is executed. The body of a detectable can contain one specific action: abort, complete, impasse or continue. The StartDepressurizing detectable specifies an abort action. The detectable says that if the agent gets a belief (either through the detection of a fact in the world, reasoning or communication) that the EVACrew agent's next subactivity is to step into the airlock, it will abort the working workframe, which means it will end the activity SuitDonning.

The actual behavior of the agent is thus dependent on which of its workframes fire and when. Firing of workframes depends on the beliefs of the agent at every moment in time. The beliefs in the belief-set of the agent depend on the initial-beliefs, conclude statements in thoughtframes and workframes that fire, communication with other agents, and detection of facts in the world. The behavior of the agents is therefore situation specific and it is not only dependent on its internal reasoning (using thoughtframes), but also determined by the interaction of the agent with other agents and with the modeled environment. We refer to this Brahms modeling paradigm as a situated activity paradigm.

Activity Subsumption Architecture

An important aspect of the Brahms activity paradigm is that activities are not the same as functions and procedures in imperative languages (Pratt & Zelkowitz, 1996). Imperative languages use a computer memory-based program stack to keep track of function calls. When a function is executed, the function's context is 'pushed onto the program stack. When in a subfunction the program is not also still in the context of the 'parent' function. Thus the program cannot move execution back and forth between a function and its subfunctions that are called. Function execution is sequential and cannot be interrupted. Not so with activities. In contrast, Brahms activities stay active while they are being executed.

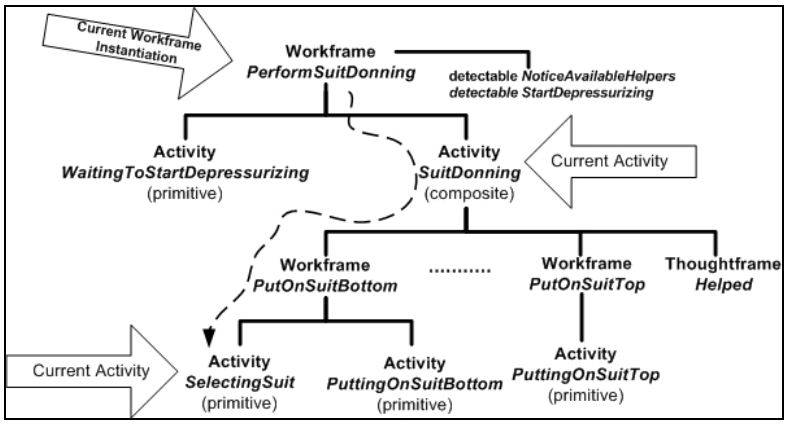


Figure 13. PerformSuitDonning Activity Subsumption

Thus, if a subactivity in a workframe of a composite activity gets executed, the 'parent' composite activity is still active (see Figure 13). All workframes, thoughtframes and detectables in the 'parent' activity are still being evaluated while the agent is executing the subactivity. This is part of the Brahms subsumption architecture (Brooks, 1991), and is based on the principle that humans are always multi-tasking by being in a hierarchy of activities at the same time. For example, the EVACrew member is also still in the activity of SuitDonning when it is in the activity of SelectingSuit. Thus, every workframe, thoughtframe or detectable in the current activity hierarchy is part of the agent's context, and can be fired at any moment, changing the belief and behavioral state of the agent. However, at the same time workframes outside of the current activity tree also have the potential to fire, enabling an activity-context switch for the agent (e.g. the workframe PutOnSuitTop in Figure 13).

In a Brahms simulation, an agent may engage in multiple activities at any given time, but only one activity in one workframe is active at any one time. At each event the simulation engine determines which workframe should be selected as the current working, based on the priorities of available, current and interrupted work (see Figure 11). The state of an interrupted or impassed workframe is saved, so that the agent can continue an interrupted workframe with the activity that it was performing at the moment it was interrupted.

An important consequence and benefit of this subsumption architecture is that all of the workframes of a model are simultaneously competing and active, and the selection of a workframe to execute is made without reference to a program stack of workframe execution history.

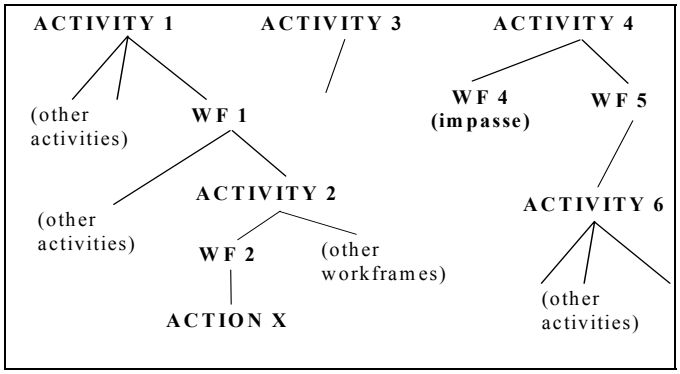


Figure 14. Multi-tasking in Brahms

An illustration of this is given in Figure 14. An agent (not shown) in a running model will have multiple competing activities in process: Activities 1, 3, and 4. Activity1 has led the agent (through workframe WF1) to begin a subactivity, Activity 2, which has led (through workframe WF2) to a primitive activity Action X. When Activity 2 is complete, WF1 will lead the agent to do other activities. Meanwhile, other workframes are competing for attention in Activity 1, Activity 3 and Activity 4. Activity 2 similarly has competing workframes. Priority rankings led this agent to follow the path to Action X, but interruptions or reevaluations may occur at any time. Activity 3 has a workframe that is potentially active, but the agent is not doing anything with respect to this activity at this time. The agent is doing Activity 4, but reached an impasse in workframe WF4 and has begun an alternative approach (or step) in workframe WF5. This produced a subactivity, Activity 6, which has several potentially active workframes, all having less priority at this time than WF2.

The Brahms subsumption architecture allows two forms of multi-tasking. The first form is inherent in the activity-based paradigm; Brahms can simulate the reactive situated behavior of humans. An agent's context forces it to be active in only one low-level activity. However, at any moment an agent can change focus and start working on another competing activity, while queuing others. Having the simulation engine switch between current and interrupted work for each agent, simulates this type of multi-tasking behavior as represented in Figure 14. The second form is subtler. People can be working concurrently on many high- and medium-level activities in a workframe-activity hierarchy. While an agent can only execute one primitive activity in the hierarchy at a time (e.g. ACTION X in Figure 14), the agent is concurrently within all the higher–level activities in the workframe-activitiy hierarchy. For example, the agent in Figure 14 is concurrently within Activity 1, Activity 2 and primitive activity Action X, Activity 3, Activity 4 and Activity 6. It should be noted that while a workframe, and its associated activities are interrupted or impassed, the agent is still considered to be in the activity. The agent is conceptually within all current, interrupted and impassed activities.

To exemplify this subtle multi-tasking behavior of an agent, we draw the reader's attention to EVAHelper agent CC's timeline in Figure 9. The agent has two simultaneously competing workframes in the PerformEvaPrep composite activity, namely workframe PerformIndividualActivitiesDuringEva and OfferHelp- WithSuitDonning. The agent starts with executing the PerformIndividual- ActivitiesDuringEva workframe. This workframe gets interrupted when agent RZ asks for help (see communication line from agent RZ to agent CC in Figure 9). At that time agent CC starts executing workframe OfferHelpWithSuitDonning. When the agent is done with this workframe the interrupted workframe becomes active again, and the agent continues executing the PerformIndividualActivitiesDuringEva workframe there where it left off before its interruption. This worframe interruption is shown in the timeline in Figure 9 with red vertical bars at the beginning and end of the workframe's interruption.

The Reasoning Component in Brahms

The reasoning component consists of production rules for agents and belief conclusions in workframes. Production rules in Brahms are forward chaining inference rules acting on the beliefs of an agent. These rules are called thoughtfames, because using these rules an agent can 'think' and deduce new beliefs while in an activity. Each agent has a set of thoughtframes, a combination of thoughtframes locally declared and inherited. Table 4 shows a thoughtframe.

Table 4. Thoughtframe

```
thoughtframe Helped {
        repeat: true;
        variables:
                forone(EVAHelper) evaHelper;
        when(
                knownval(current.currentHelper = evaHelper) and
                knownval(evaHelper.helping = true) and
                knownval(current.needHelp = true) and
                knownval(current.beingHelped = false))
        do {
                conclude((current.beingHelped = true));
        }
}
```

A thoughtframe consists of a number of elements which we will describe using Table 4. First of all, a thoughtframe is used to infer new beliefs based on current beliefs in the belief-set of the agent. New beliefs are created when a thoughtframe executes. The conclude statement in the do-part or body of the thoughtframe creates a new belief for the agent. Table 4 shows a conclude statement of the form (O.A = ν), where O = 'current', A = [the 'beingHelped' boolean attribute of the agent] and ν is the outcome of a boolean expression that is evaluated before the belief is created (ν= true).

When the agent has one or more beliefs that are matching all the preconditions the thoughtframe is immediately executed. Using this approach we can represent the forward-reasoning behavior of an agent; the conclude sttement in one thoughtframe can trigger the execution of a subsequent thoughtframe or workframe, thus creating a 'forward chaining' of belief-set changes simulating the reasoning behavior of a person. Every time the agent gets a new belief, only those thoughtframes are evaluated that have a precondition that is a potential match on the newly created belief. This makes the reasoning behavior efficient, because for every belief change event in an agent only a small number of preconditions have to be evaluated.

The activity and reasoning components in Brahms are integrated in a way that allows the modeler to represent reasoning in the context of an activity. This situated reasoning is accomplished with composite activities. Composite activities consist of both workframes and thoughtframes. Figure 13 shows how the Helped thoughtframe is part of the SuitDonning composite activity. The agent can only execute the thoughtframe when performing the composite activity. Worframes can also create new beliefs (and facts) in the world (using a conclude statement). This allows the modeler to create belief changes and world-state changes (facts) of the agent, representing consequences of activity execution in a workframe. In other words, situated reasoning of an agent is done in context of the current activity and can be represented both as a consequence of the activity (as belief-consequences in workframes) and as a deliberative reasoning process during the activity (as thoughtframes).

3.2. The Animation Component—BrahmsVE

The animation component presents the Brahms agents and objects in a 3D virtual world. Just as the reasoning and behavioral components, the animation component is a model that is developed based on the scenario description. However, importantly, the animation component needs to be developed in close connection with what is in the behavioral model. The activity behaviors represented in the Brahms model need to be animated via the animation component. The Brahms agents and objects need to have an embodiment as bots inside the virtual world. With embodiment comes the notion of a geography model that needs to correspond with a cluster in the virtual world. A cluster consists of a collection of locales, and is a 3D model that corresponds to a Brahms Geography model. A locale is developed based on an animation storyboard created from the scenario description and the photo and/or video data available. The development cycle of all these models is represent in Figure 15. The next sections describe how these models are developed.

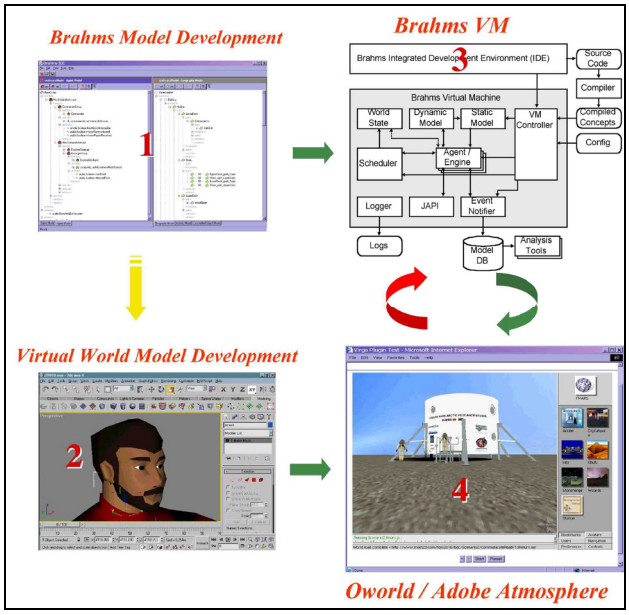


Figure 15. BrahmsVE Modeling Cycle

The Geography Model

The Geography Model specifies the location where agents perform activities and objects can be found. The Geography Model consists of a set of areas representing confined actual or conceptual spaces where agents and objects can be located. Areas can be contained inside other areas forming a conceptual hierarchical containment definition of space in the real-world. Area hierarchies enable the representation of spaces such as the FMARS habitat, containing floors that contain rooms and specific locations inside a room. Figure 16 shows the area hierarchy for the EVA preparation room on the lower deck of the FMARS habitat where the spacesuit donning activity is performed.

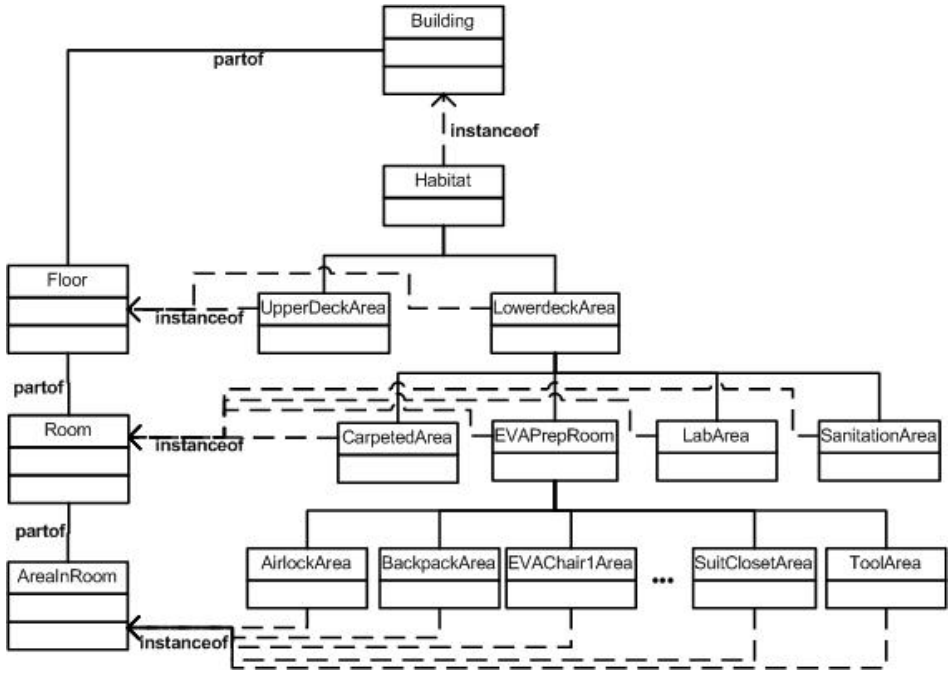


Figure 16. Habitat Geographical Area Hierarchy

The Brahms modeler is responsible for specifying the geography and placing agents and objects in the geography in their initial locations at the start of the scenario (see the agent inhabitants BC, CC, KQ, RZ and VP in Figure 17). The geography is to be visualized in a three-dimensional view by OWorld. OWorld displays the initial geography or scene, and is the virtual reality environment (see Figure 7) that consists of components that can visualize a Brahms geography in a three-dimensional view and can visualize the activity movement and behaviors of agents and objects in this view.

The Brahms geography model consists of Brahms source code not consisting of any specifics on how the geography, agents and objects are to be visualized in a three-dimensional view. The Brahms developer uses the Composer to design the geography and position the agents and objects within the geography. The Composer stores this information in a XML format parsable by Oworld (see Figure 17). The file contains information on how to organize and visualize the areas, agents and objects and where to position them.

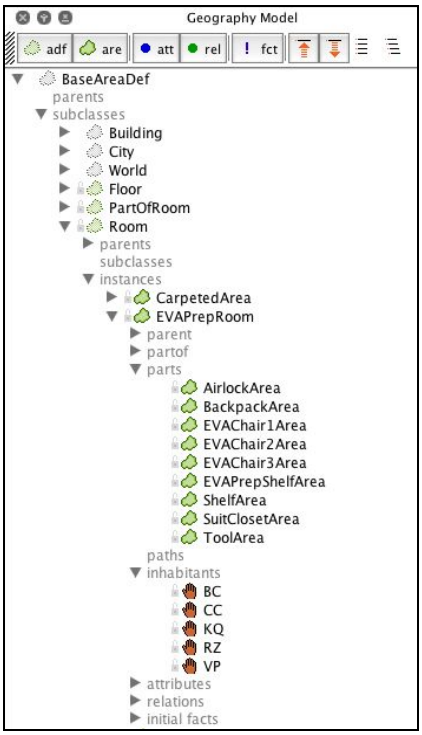


Figure 17. EVAPrepRoom Area Geography Model

The Virtual World Model—OWorld

Areas are worlds or locations in a world that can represent various types of living spaces, possibly organized in a containment hierarchy. Agents and objects are entities that have location, can move around and can interact with each other. Some of the areas in Brahms models are grouped into more high-level areas, where a number of sub-areas are located fairly closely together and in which agents and objects are confined to the high-level area. A high-level area like this has its own three-dimensional visualization and is called a Locale. An example of a locale is the lower deck of the FMARS habitat shown in Figure 18, first as a drawing, then as actual photographs from within the FMARS habitat and finally as a 3D locale inside Adobe Atmosphere. This locale is represented in Brahms as shown in Figure 16 and Figure 17. The collection of locales with their visualizations and positioning of areas, agents and objects is called a cluster. An example of a cluster is the complete geography model for the EVAPrep model. OWorld does not render a cluster all at once, but renders locales and a user generally views only one locale at a time.

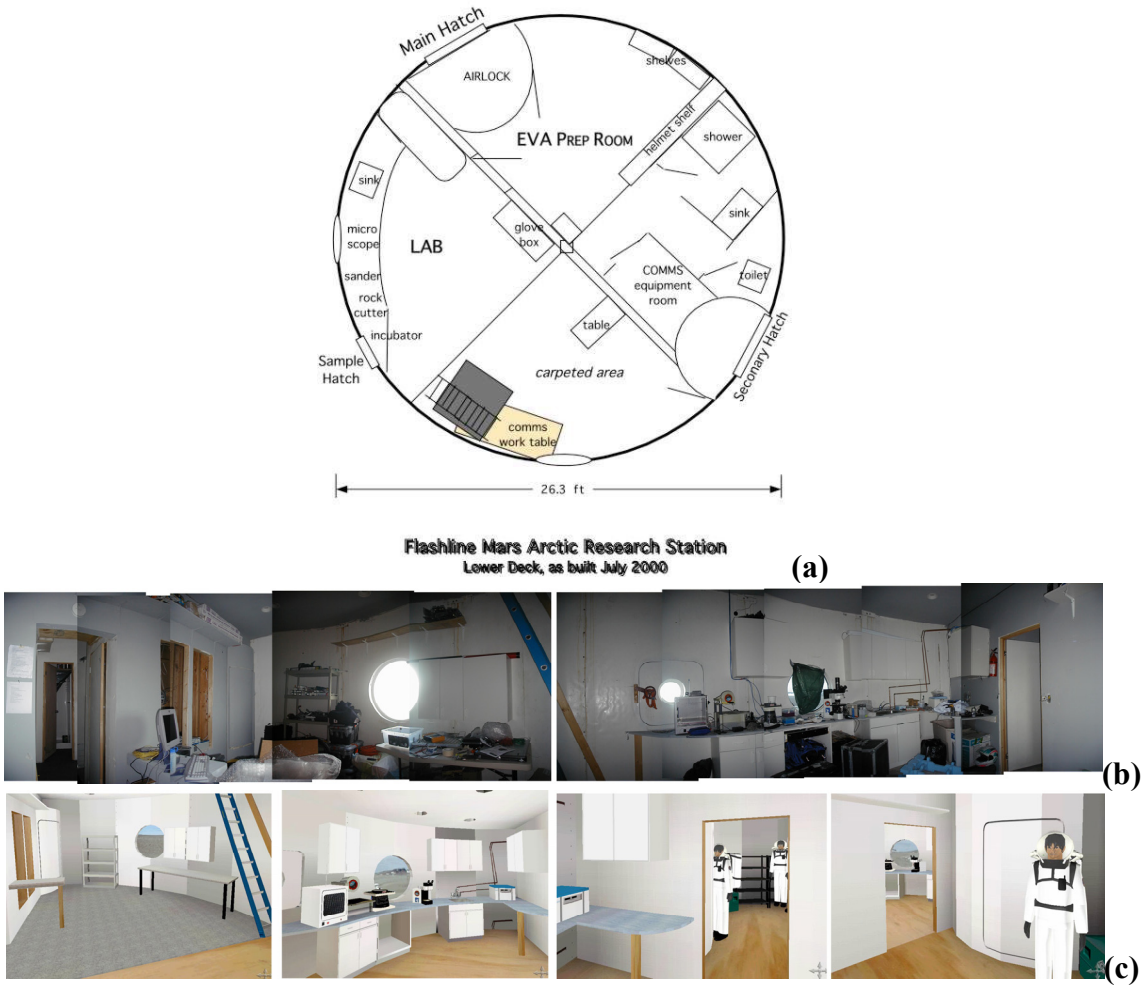


Figure 18. (a) FMARS Lower deck Drawing (b) actual carpeted + lab areas (c) locale of Lowerdeck

In order for OWorld to render a locale, it needs to know how and where to render the various Brahms concepts. This information is specified using XML. OWorld loads in the cluster XML file and can render the default locale using the information specified in the file. OWorld will represent agents and objects as virtual bots. Areas will be represented as points. A point is a labeled location in a locale (a sub-area in the Brahms model). A point represents one specific X, Y, Z coordinate and is used to represent a named Brahms areas—parts—in a locale. The scene graph will have an accurate representation of the area and its size (see Figure 18c showing the carpeted-, lab- and EVA prep room). The point that represents the area merely serves as a point for agent movement used during a simulation to map an area to a location in OWorld. Brahms agents and objects move from one area to another without the use of a coordinate system, while OWorld moves the agent and object bots according to the VW coordinate system. The area to point mapping allows the generation of movement in the VW renderer (Adobe Atmosphere or Ogre). How this is actually done is described in the next section.

OWorld Events and Activity Animation

There are currently three types of events that can be used to visualize a Brahms model simulation in OWorld. These are described in Table 5.

Table 5. OWorld Event Types

Event Type	Description	Format
setVisible	A bot or icon can be set to be visible or invisible by using the setVisible event.	setVisible|time|bot|visible
activity	A Brahms agent can execute an activity via its virtual representation (bot). This event is used to have the bot execute an animation script animating the activity of the bot in the virtual world. Examples are getting up, walking, standing somewhere, opening a door, talking to someone, etc.	activity|activitytype|starttime|endtime|who|activityname|arguments
setLabel	The label of a bot can be changed by using the setLabel action.	setLabel|starttime|endtime|labeltype|displaytext|font|fontsize|colorR|colorG|colorB

Agent activities that are simulated in the Brahms VM are recorded as OWorld activity events and rendered in the VW by OWorld. For example, during the SuitDonning composite activity agents RZ and CC their activities (Figure 19) generate the activity events in Table 6 (only showing the events for the first half of the Figure 19).

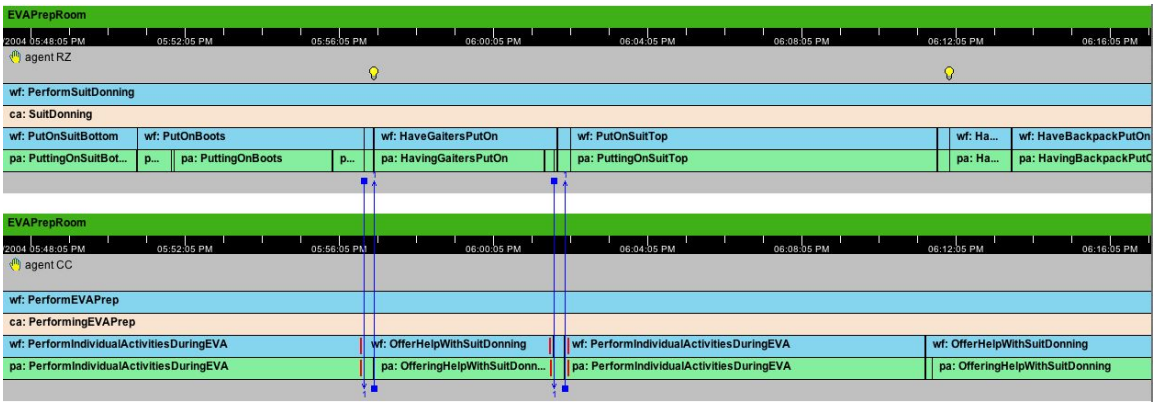


Figure 19. Agent RZ executing the CheckingBeforeDonning composite activity

Table 6. Brahms-OWorld events generated from the CheckingBeforeDonning activity

activity|**primitive**|803|1044|projects.EVAPrep.**RZ**|**PuttingOnSuitBottom**|

activity|**get**|1044|1088|projects.EVAPrep.**RZ**|**SelectingBoots**||projects.EVAPrep.BootsRZ|projects.EVAPrep.ShelfArea

activity|**primitive**|1088|1093|projects.EVAPrep.**RZ**|**SittingDown**|

activity|**primitive**|1093|1358|projects.EVAPrep.**RZ**|**PuttingOnBoots**|

activity|**primitive**|0|1397|projects.EVAPrep.**CC**|**PerformIndividualActivitiesDuringEVA**|

activity|**get**|1358|1405|projects.EVAPrep.**RZ**|**SelectingGaiters**||projects.EVAPrep.GaitersRZ|projects.EVAPrep.ShelfArea

activity|**primitive**|1397|1414|projects.EVAPrep.**CC**|**CommmunicatingReadinessToHelp**|

activity|**primitive**|1405|1417|projects.EVAPrep.**RZ**|**CallingAvailableHelper**|

activity|**primitive**|1417|1667|projects.EVAPrep.**RZ**|**HavingGaitersPutOn**|

activity|**primitive**|1414|1689|projects.EVAPrep.**CC**|**OfferingHelpWithSuitDonning**|

… etc.

The events entering OWorld are used to generate the animation of the bots in the VW, representing the agent's activities in the Brahms simulation. Animations are done using the Java-script scripting language. A script associated with the OWorld bot-class representing the agent animates each activity event. For example, agent movement is animated with the actionHumanWalk function shown in Table 7.

Table 7. Java-script code-fragment for animating an agent's move activity

```
actionHumanWalk.prototype.start=function(args)
{
        …
        // Calculate the total length and use it to calculate the
        // total time the animation will take.
        this.totaltime = fTime;
        // Notify the sync server of the new position
        sendBrahmsString("interface|setSettable|" +this.owner.getTime()+ "|" +this.owner.m_name+
"|Location|"+ args[4].m_name );
        // Start up the walking animation
        this.owner.contAction.setTime( "walk", 0);
        this.owner.contAction.setWeightPercTarget( "walk", 1, 1 ); return this.totaltime;
}
```

The combination of the Brahms model of agent activities, communications and interactions, and the simulation with event output to Oworld driving the OWorld Java-script execution, generates bot animation of the agent's activities in the virtual world. Figure 20 shows screenshots of the bot animation of agent RZ and CC donning their space suit, corresponding the activities of the agent in the simulation model—the reader should compare the screenshots from Figure 20 with the agent's activity timeline in Figure 19.

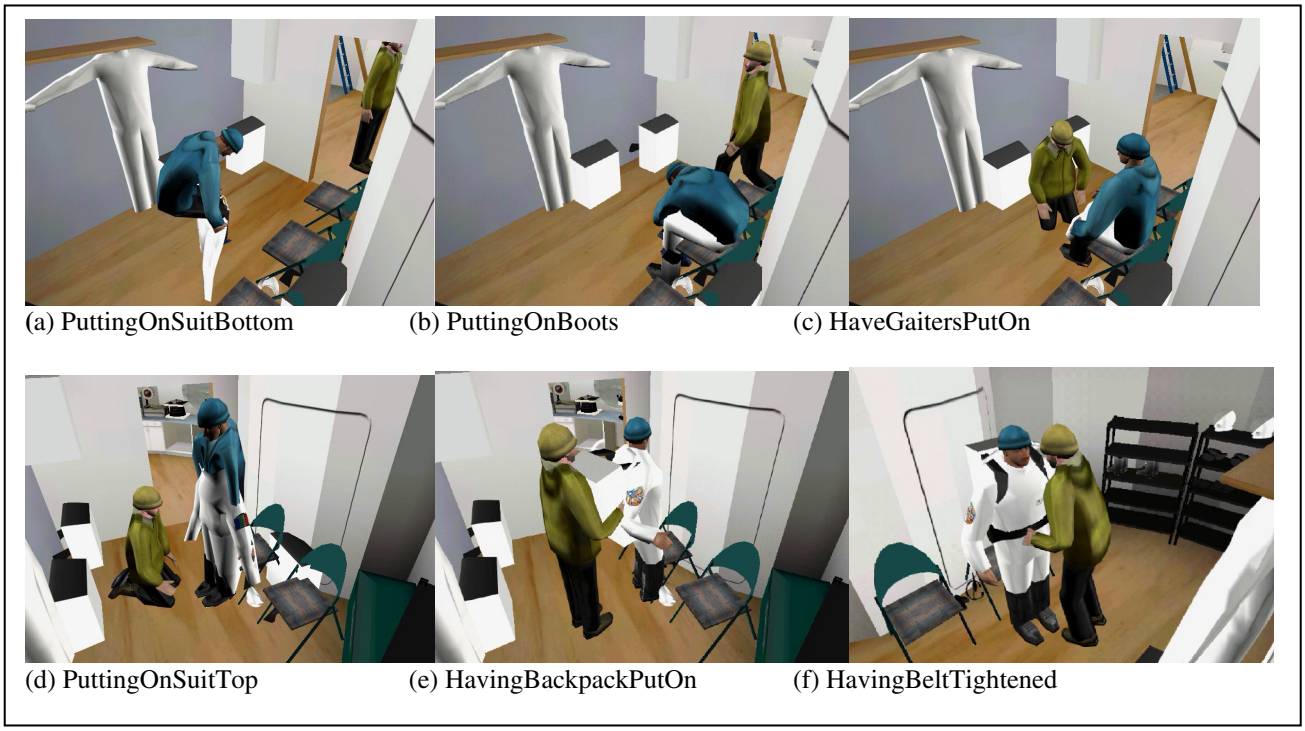


Figure 20. DonningSpaceuit Animation

4. Discussion: Ongoing and Future Work

The discussion thus far explained how the Brahms model simulation "drives" the animation of bots in the virtual world. Although important, this is only part of the story. To enable a true interactive virtual world, one in which the user can interact with the bots through their avatar, there needs to be a two-way communication between the bot's activity model in Brahms and the virtual world. In other words, the fact that the user can enter the VW means that the world is not stable, but dynamic. Users' avatars can enter and leave the world, changing it by adding and deleting objects in the world and interacting with bots in a way that was not foreseen by the bot and VW modelers. At the same time, animated bots are the embodiment of the agents in a 3D virtual world. The agent's embodiment as a 3D-character in the virtual world constrains it in its capability to act, just as people are constraint in their actions in the real world by their environment. For example, just as people cannot walk through walls and other objects, neither can bots in the VW. This embodiment in a 3D virtual world enables the agent to get additional sensory information from it. For example, line of sight is an important constraint on what the agent can observe at any moment in time. In other words, activities of agents are constrained by the bot's embodiment and sensory-input from the 3D world. The distinction between an agent's "mind and body" disappears as we implement a tight coupling between the two. This coupling between cognition, motor-skills and sensory input instantiates our notion of situated activities. The 3D virtual world enables and constrains the agent's activity behavior at the same time. Madsen and Granum describe the three-way interaction, between their VE server, low-level and high-level agent, as high-level sensory information, or percepts, (Madsen & Granum, 2001). We

are currently working on including this capability and in this last section we discuss how we are implementing this in the BrahmsVE environment.

Agent perception of the Virtual World: An integration of the VW and the agent behavior model

As mentioned above, representing and executing agent behavior is only one of the challenges we face when dealing with intelligent behavior in virtual worlds. A more difficult challenge is to integrate agent behavior within an always-changing virtual world. Some of these challenges are:

- How does an agent "see" things (objects and other agents) in the VW?
- How much and how far away can an agent "see" at any given moment?
- How does an agent know what it sees in the VW?
- How can we simulate sound in the VW?
- How does an agent "hear" things in the VW?
- How much and how far away can an agent "hear" at any given moment?
- How does an agent detect collisions with ("feel") other agents and objects?
- How does an agent know what is in its vicinity?

We deal with these challenges by separating the capabilities needed for agent perception between generation of the phenomena, detection of the phenomena and agent reaction to the detected phenomena. The first two capabilities are implemented in OWorld, while the third capability is implemented with the standard Brahms concepts of detectables, thoughtframes, workframes and activities.

Events that can impact Brahms agent behavior:

1. End move: A Brahms agent moves from one area to another by executing a move activity. Traditionally, the Brahms VM determines how long the move takes, based either on a pre-specified activity time, or the calculation of the path, based on a shortest route algorithm. However, being within a 3D VW the actual path of the move is determined by the possible paths through the VW (doors, objects in the way, et cetera). Therefore, ending Brahms agents' move activities is determined by OWorld, and the agent has to wait for a signal from OWorld that the bot arrived at its end location. Thus, OWorld will send an end-move event to the Brahms VM, which in turn has to react appropriately. Using this approach, one issue is; What happens if a move cannot be completed and an agent in OWorld is still en route? Brahms has no coordinate system, just areas. The agent might stop moving at a point in the VW that does not correspond with an existing area in the Brahms geography model. Should the Brahms VM send the agent back to the original location? Or more appropriately, should the bot stay at the location of the interrupted move, and "release" control back to the Brahms VM letting the agent model decide what to do next? It is now up to the Brahms modeler to include agent behavior that enables the agent to react appropriately. In this case, the issue is how to deal with the fact that a bot can stop anywhere in the VW and thus at locations in a locale that does not have a corresponding area in the Brahms model. In that case, how does the agent know where it is? One way to solve this problem is for OWorld to dynamically create a new area in the Brahms VM, and place the agent there. Using this approach the Brahms VM will handle the belief creations for the agent and the agent will now know where it is.

2. End activity: Activities have gestures associated with them that cost real time to animate in OWorld. In other words, the time to complete an activity is determined by the time it takes to animate the activity for the bot in the virtual world. The Brahms VM will have to wait for OWorld to tell it that the activity has indeed been completed. One way to deal with this is that the Brahms VM will send OWorld the "desired" end-time for the activity. OWorld will use this time to generate the animation duration and thus be able to determine the "speed" of the animation. As in the end move activity determination, what should happen when the agent in Brahms is interrupted while in its current activity before OWorld completed its animation? Stopping the animation "in the middle" is difficult, given the fact that the animation is generated using a script. At the moment, we are still working on finding an appropriate solution to this issue.

3. Collision events based on auras: "Seeing" and "hearing" of an agent are implemented with the notion of an aura. Figure 21 shows the vision and auditory aura of two bots, represented by the two small black circles. The auditory aura is a 360^{o} circle, allowing the bot to "hear" sounds from any direction. The vision aura is an elliptic area in front of the bot, limiting the bot to only see what is in front of it. Figure 21 shows that both the vision and auditory auras of the two bots overlap.

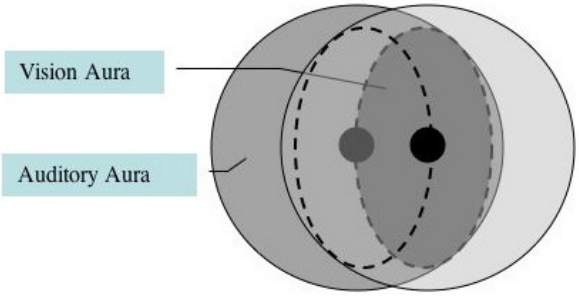


Figure 21. Vision and Auditory Auras of two bots

If the auras of agents overlap a collision event is send to the Brahms VM. Brahms then generates an appropriate fact ("hearing" a sound, and "seeing" an object or another agent in the VW). Such facts can then be detected by the agents using the standard Brahms detectables, allowing the agents to react appropriately. For example, an agent can interrupt a move to initiate a "conversation" with an agent it "sees". However, this could make the animation choppy, because of the time needed to do the collision processing. Should OWorld wait for Brahms to send a 'collision processed' event back to OWorld for each interruption, so that it can then continue with the appropriate animation? If not, a moving bot might be "out of sight" before the collision response takes effect. Should Brahms indicate to OWorld what types of collisions it is interested in for an activity, i.e. whenever a detectable becomes active the VM notifies OWorld of that detectable and through some mapping it can be linked to types of collisions? This optimizes the collision processing. If an agent is never interested in talking to somebody during a certain activity, it does not need to be bothered with collision detection of auras overlapping.

4. Field of vision (the vision aura): All objects that are visible by the agent in OWorld should be "communicated" to the Brahms agent. Brahms' current auto-detection mechanism should be shut off (normally used when entering an area). OWorld completely drives what an agent sees and therefore implicitly detects. Again, should the Brahms agent notify OWorld of the objects it is

interested in, making object detection explicit in the model and thus more efficient? Doing this would mean that the agent "decides", explicitly, which objects it will "see". This does not represent real life, since people do not consciously decide, before entering an area, what objects they will notice.

5. How do we map new event info from OWorld to Brahms? We ensure that whatever is in the VW needs a representation in the Brahms model, so that their identifiers can be used for the mapping? This means every object, agent and user avatar in the VW requires a corresponding representation in the Brahms model. An agent cannot detect (i.e. "see" or "hear") an object, bot or avatar that does not have a corresponding representation in the Brahms model. This allows a user entering the VW as an avatar to either be detected by the agents or not. The avatar cannot be detected as long as a corresponding agent is not created inside the Brahms VM.

Displaying mental state in the Virtual World

How can the end user understand what a bot is "thinking" about? This problem we had not foreseen until our first experiment with virtual worlds. For training applications it is important that the end-user (i.e. the trainee or trainer) gets a good understanding of what is happening in the training session. Without the ability to immediately understand what the bots are doing and "thinking," understanding the situation is sometimes very difficult. To alleviate this otherwise inevitable situation, we devised a way for the agents to communicate mental state to the end-user. First off, we allow the agents to speak to the end-user by generating text-to-speech. However, with a lot of bots within the user's view it can become difficult to know which bot is speaking. To solve this problem, BrahmsVE allows the display of a text-balloon next to the bot that is speaking. This way both text and speech are observed. A more difficult problem is to communicate an activity's relevant mental state of an agent to the end-user. Figure 22 shows how we communicate to the end-user that the Personal Satellite Assistant (PSA) agent has found the drill it was asked to look for?

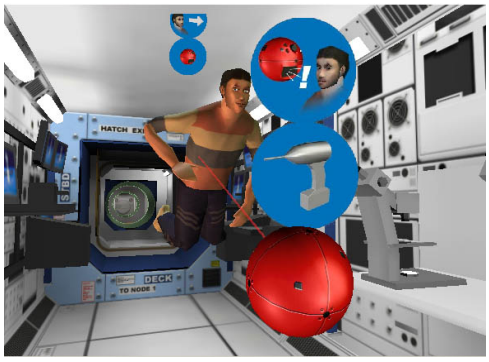
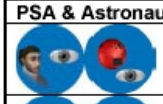


Figure 22. PSA reports the location of the drill, while the astronaut looks at the PSA

We devised a number of status icons that communicate the mental state of an agent to the end-user. Throughout the simulation the mental status of the agents is reported in a text buffer and indicated by convenient status icons projected above the active agents. The visual language for these status icons is described in Table 8 and Table 9.

An activity icon is an indicator of the activity the agent is currently performing. Table 8 shows the activity icons that were developed for the International Space Station (ISS) model, in which an astronaut agent asks a PSA agent to locate a drill in the ISS.

Table 8. Activity Icons (top icon)

PSA & Astronauts	
	The Agent is looking at, and tracking, the Target.
	The Agent is moving to a new location.
PSA Only	
	The PSA is currently scanning its surroundings for its Target.
Power Lead	The PSA is currently recharging.
	The PSA is reporting the tool' location.

The target icon represents what object the agent is currently focused on. Table 9 shows the target icons that were developed for the ISS model. Status icons are made visible by a setVisible event from the BrahmsVM to OWorld.

Table 9. Target Icon (bottom icon)

Icons	Target
	Drill, Flashlight or Wrench tools
	PSA is receiving commands (via laptop)
	Agent sees PSA or Astronaut
	PSA is affected by blower fan
	PSA has identified and is utilizing power station

We are currently implementing both the agent virtual world perception capabilities mentioned above, as well as the mental state display capabilities. The BrahmsVE will allow us to develop rich virtual world environments in which bots are not just simple script-driven bots, but can animate complex activity behavior and reasoning and interact with the end-user. We are at the very beginning of enabling the development of just-in-time training applications for space missions. BrahmsVE will allow us to experiment with richer virtual worlds in which end-users can start to interact with complex behavioral bots via their avatars. The Brahms language is a previously tested rich multi-agent language for modeling complex bot-behaviors, and by integrating this language with OWorld we can enable the development of complex virtual worlds on the web. As the next step, after the BrahmsVE environment is sufficiently complete, we will start our research of how just-in-time mission training applications could and should be developed.

Acknowledgements

The work described in this chapter was done from 1999-2002, in close collaboration between the Brahms team at NASA Ames Research Center and DigitalSpace Corporation. DigitalSpace received its funding for this work from a Small Business Technology Transfer (STTR) grant and two Small Business Innovation Research (SBIR) grants. The Brahms team received funding for this work from NASA's Computing, Information and Communications Technology Program (CICT), in the Human-Centered Systems area of the Intelligent Systems (IS) project.

References

ActiveWorlds. (2004). http://www.activewords.com. Activeworlds Corporation [2004, August 28].

Atmosphere. (2004). http://www.adobe.com/products/atmosphere/main.html. Adobe [2004, August 28].

Blumberg, B. (1996). Old Tricks, New Dogs: Ethology and Interactive Creatures. Unpublished PhD thesis, Massachusetts Institute of Technology, Cambridge, MA.

Brooks, R., A. (1991). Intelligence without representation. Artificial Intelligence, 47, 139-159.

Brooks, R. A. (1997). From Earwigs to Humans. Robotics and Autonomous Systems, Vol. 20(2-4), 291-304.

Clancey, W. J. (2001). Field Science Ethnography: Methods for Systematic Observation on an Expedition. Field Methods, 13(3), p. 223-243.

Clancey, W. J. (2002). Simulating "Mars on Earth"—A Report from FMARS Phase 2. In F. Crossman & R. Zubrin (Eds.), On to Mars: Colonizing a New World: Apogee Books.

Clancey, W. J. (2002). Simulating Activities: Relating Motives, Deliberation, and Attentive Coordination. Cognitive Systems Research, 3(3), 471-499.

Clancey, W. J., Sachs, P., Sierhuis, M., & van Hoof, R. (1998). Brahms: Simulating practice for work systems design. International Journal on Human-Computer Studies, 49, 831-865.

Clancey, W. J., Sierhuis, M., Damer, B., & Brodsky, B. (in press). Cognitive modeling of social behaviors. In R. Sun (Ed.), Cognition and Multi-Agent Interaction: From Cognitive Modeling to Social Simulation. New York: Cambridge University Press.

d'Inverno, M., Luck, M., Georgeff, M., Kinny, D., & Woodbridge, M. (2004). The dMARS Architecture: A specification of the Distributed Multi-Agent Reasoning System. Autonomous Agents and Multi-Agents Systems, Kluwer Academic Press, Vol. 9(1/2), 5-53.

Damer, B. (1997). Avatars! : Exploring and Building Virtual Worlds on the Internet. Berkeley, CA: Peachpit Press.

Huber, M. J. (1999). JAM: A BDI-theoretic mobile agent architecture. Paper presented at the Proceedings of the Third International Conference on Autonomous Agents (Agents'99), Seattle, WA.

Lave, J. (1988). Cognition in Practice. Cambridge, UK: Cambridge University Press.

Madsen, C. B., & Granum, E. (2001). Aspects of Interactive Autonomy and Perception. In L. Qvortrup (Ed.), Virtual Interaction: Interaction in Virtual Inhabited 3D Worlds (pp. 182-208). London: Springer Verlag.

Madsen, C. B., Pirjanian, P., & Granum, E. (1999). Can finite state automata, numeric mood parameters and reactive behavior come alive? Paper presented at the Proceedings of the Workshop on Behaviour Planning for Life-Like Characters and Avatars, Spain.

Ogre. (2004). http://www.ogre3d.org/. Ogre3d [2004, 11/29/2004].
Pratt, T. N., & Zelkowitz, M. Z. (1996). Programming Languages: Design and Implementation (3rd. ed.). Englewood Cliffs, NJ.: Prentice Hall.
Schön, D. A. (1982). The Reflective Practitioner: How Professionals Think in Action: Basic Books.
Sierhuis, M. (2001). Modeling and Simulating Work Practice; Brahms: A multiagent modeling and simulation language for work system analysis and design. Amsterdam, The Netherlands: University of Amsterdam, SIKS Dissertation Series No. 2001-10.
Sierhuis, M., Clancey, W. J., & Hoof, R. v. (Submitted). Brahms: A multiagent modeling environment for simulating work practice in organizations. Journal for Simulation Modelling Practice and Theory, Elsevier, The Netherlands, Special issue on Simulating Organisational Processes.
Suchman, L. A. (1987). Plans and Situated Action: The Problem of Human Machine Communication. Cambridge, MA: Cambridge University Press.
van Hoof, R., & Sierhuis, M. (2000). Brahms Language Reference. http://www.agentisolutions.com/documentation/language/ls_title.htm. Available: http://www.agentisolutions.com/documentation/language/ls_title.htm.

Chapter 9

Pavlovian, Skinner, and Other Behaviourists' Contributions to AI

***

Witold Kosinski and Dominika Zaczek-Chrzanowska
Polish-Japanese Institute of Information Technology, Research Center
Polsko-Japo ska Wy sza Szkoła Technik Komputerowych
ul. Koszykowa 86, 02-008 Warszawa
wkos@pjwstk.edu.pl mado@pjwstk.edu.pl

Abstract

A version of the definition of intelligent behaviour will be supplied in the context of real and artificial systems. Short presentation of principles of learning, starting with Pavlovian's classical conditioning through reinforced response and operant conditioning of Thorndike and Skinner and finishing with cognitive learning of Tolman and Bandura will be given. The most important figures within behaviourism, especially those with contribution to AI, will be described. Some tools of artificial intelligence that act according to those principles will be presented. An attempt will be made to show when some simple rules for behaviour modifications can lead to a complex intelligent behaviour.

1. Intelligence: Description

It can be stated without any doubt that behaviourists have made a great contribution to the development of artificial intelligence. The evidence from the animal learning theory, especially the laws of learning discovered by behaviourists, has attracted researchers within artificial intelligence for many years and many models have been based on them.

Intelligence is a complex and controversial concept, therefore it is very difficult to capture it by a simple definition. According to Jordan and Jordan [1] it is appropriate to regard intelligence as a concept we employ to describe actions of a certain quality. Two criteria should be used in this regard, namely, speed (i.e. how quickly an agent performs a particular task requiring mental ability) and power (i.e. the degree of difficulty of the tasks an agent can perform). On the other hand one can find another definition of intelligence expressed in term of an ability to perform cognitive processes. There are three fundamental cognitive processes: 1) abstraction, 2) learning, and 3) dealing with novelty.

Intelligence has been given many definitions by prominent researchers in the field, for example, it has been defined as:

- A general ability which involves mainly the education of relations and correlates. (Spearman, 1904) [2]
- The ability to judge well, to understand well, to reason well. (Binet and Simon, 1905) [3]
- The capacity to form concepts and to grasp their significance. (Terman, 1916) [4]
- The ability of individual to adapt adequately to relatively new situations in life. (Pintner, 1921) [5]
- The power of good responses from the point of view of truth or fact. (Thurstone, 1921) [6]
- The mental capacity to automatize information processing and to emit contextually appropriate behaviour in response to novelty; intelligence also includes metacomponents, performance components, and knowledge-acquisition components. (Sternberg, 1986) [7]

There are two main approaches to describing intelligence: the psychometric approach and the information-processing. The psychometric approach focuses on measuring or quantifying cognitive factors or abilities that make up an intellectual performance. Those cognitive factors might include: verbal comprehension, memory ability, perceptual speed, and reasoning. The scholars who follow this approach either lump these factors together (lumpers) or split them apart (splitters).

According to lumpers, intelligence involves a general unified capacity for reasoning, acquiring knowledge, and solving problems. The most well-known theory is Spearman's two-factor theory [2]. Spearman proposed that intelligence consisted of two factors: a single general factor (*g*) and numerous specific factors (*s*). The performance in any test or task is a function of both *g* and *s*. The idea of general intelligence factors is behind using a single measure of intelligence, such as an IQ (intelligence quotient) score.

In contrast to lumpers, splitters define intelligence as composed of many separate mental abilities that function more or less independently. According to well know Gardner's multiple-factor theory [8], there are at least seven independent aspect of intelligence: verbal skills, math skills, spatial skills, movement skills, musical skills, insight about oneself, and insight about others. Gardner stated that understanding these aspects comes from studying person in his or her environment and not from results of IQ tests.

The competitive approach to intelligence—information-processing approach—defines intelligence by analyzing the components of the cognitive processes that people use to solve problems. The well-known example of this approach is Sternberg's triarchic theory [7]. Sternberg proposes that intelligence can be divided into three ways of gathering and processing information. The first uses analytical or logical thinking skills that are measured by traditional intelligence tests. The second uses problem-solving skills that require creative thinking, the ability to deal with novel situations, and the ability to learn from experience. The third is using practical thinking skills that help a person to adjust to and to cope with his or her sociocultural environment.

Although experts provide us with many definitions of intelligence they tend to agree that intelligence is: (1) the capacity to learn from experience and (2) the capacity to adapt to one's environment.

Not only defining intelligence but also its measurement it is a very controversial topic. The first systematic attempt to measure intelligence was made in the beginning of this century by Alfred Binet. Binet-Simon Intelligence Scale [3] contained questions that measured vocabulary, memory, common knowledge, and other cognitive abilities. Binet also introduced the concept of mental age, which became the base for computing intelligence quotient (IQ). According to him, mental age is a method of estimating a child's intellectual progress by comparing the child's score on an intelligence test to the scores of average children of the same age. Several years later Terman [4] proposed a formula to calculate IQ score. Intelligence quotient is computed by dividing a child's mental age (MA), as measured by an intelligence test, by the child's chronological age (CA) and multiplying the result by 100. Nowadays, the most widely used IQ test for adult are: Stanford-Binet tests and Wechsler Adult Intelligence Scale-Revised (WAIS-R) [9].

2. Artificial Intelligence

Traditional AI and cognitive science proceed by developing computer models of mental, human-like, functions. As a consequence, intelligence in these disciplines is closely tied to computers, it can be understood in terms of computer programs. When input is provided, input is processed, and finally output is generated. Then by analogy the human brain is viewed in some sense as a very powerful computer, as a seat of intelligence (Pfeifer & Scheier, 1999), [10]. However, when researchers in AI started applying these ideas to build robots that interact with real world, they found that it was rather difficult to have robots doing good jobs with this view of intelligence.

There are several frequent criticisms of classical AI:

- Classical AI systems *lack generalization capabilities:* Complete systems cannot be made from studies of isolated modulus.
- Classical AI systems *lack robustness* and cannot perform in real time, and run on sequential machines.
- Classical AI systems are *goal based* and organized *hierarchically;* their processing is done *centrally.*
- The real world differs from *virtual ones:* It has its own dynamics. The virtual world used in A.I. systems has states with *complete information* on them, they are *static.*
- *The frame problem appears,* i.e. how can models of parts of the real world *be kept in tune* with the real world as it is changing, and how can systems *determine* which changes in the world are *relevant to* a given situation without having to test all possible changes.

Many started looking for alternatives, and it was R. Brooks from MIT [11] who maintained that all of AI's ideas concerning thinking, logic, and problem solving were based on assumptions that come from our own introspection, from how we see ourselves. We have to focus on the interaction with the real world: Intelligent behaviour could be achieved using a large number of loosely coupled processes that function predominantly in an asynchronous, rather parallel way. This was an origin of his *subsumption architecture.* He called his new paradigm in the study of intelligence "behaviour-based robotics." (cf. Arkin, 1998) [12]. Now one often refers to the field as embodied cognitive science. Subsumption is a method of decomposing a robot's control architecture into a set of task-achieving behaviours or competences. One should add at this point

that the term behaviour is used here in two ways, in the first, more informal use, behaviour is the result of a system-environment interaction, while in the second, more technical sense, behaviour refers to internal structures, i.e. the particular layers of modules designed to generate particular behaviours (in the first sense).

In the classical AI's approach control architecture for mobile robots is functional decomposition. First information from different sensor systems is received and integrated into a central representation. Then internal processing takes place in which an environment model (world model) is built or updated together with planning of the next actions. (Here decisions concerning further actions are made.) The final stage is execution of some actions. Altogether such an appraisal leads to the *sense-think-act cycle* and the thinking act is split into a modelling and a planning activity.

In the behaviour-based robotics the main role is played by a method of decomposing a control system of a robot into a *set task-achieving behaviours* (or *competences*). This was called by R. Brook the subsumption architecture, in which control architecture is build by incremental adding competences on top of each others. This method is contrasted with the classical AI's functional decomposition. Implementations of such task-achieving behaviours are called layers: higher-level layers build and rely on lower-level ones; instead of a single information process from perception to world modelling and action, there are multiply paths, the layers that are active in parallel. A series of small subtasks of the robot's overall task are not controlled in a hierarchical, traditional way, since each layer can function relatively independently; the subsumption approach realizes the direct coupling between sensors and actuators, with only limited internal processing. In this way a direct influence of young behaviourists approach is manifested.

In a modern encyclopaedia one can read that AI is an interdisciplinary field combining research and theory from cognitive psychology and computer sciences, and which is focused on the development of artificial system that display human-like thinking or "intelligence." In other references AI is understood as any synthetic intelligence, i.e. the goal of the field of study in the above-described interdisciplinary domain. We will omit the term AI and use rather embodied AI to underline this new point of view.

3. Behaviourism

Behaviourism is considered as one of the major schools of thought in the history of psychology. This approach emphasizes the objective, scientific analysis of observable behaviours to the exclusion of consideration of unobservable mental processes. It studies how organisms learn new behaviours and change or modify existing behaviours in response to influence from their environments. The basic assumption of behaviourism is that learning is the most important factor in the development of human behaviour and the formation of personality. According to behaviourists learning is based on association between stimulus (S) and response (R) to it.

John Watson (1878-1958) is usually regarded as the father of behaviourism. According to him psychology is a purely objective experimental branch of natural science. Its theoretical goal is the prediction and control of behaviour. His ideas, published in 1913 in paper titled "Psychology as a Behaviourist Views It" [13] marked the beginning of the behavioural approach in psychology.

Although the founding of behaviourism is usually linked with the name of John Watson, many of the basic principles had already been published before Watson's time by a group of Russian researchers, in particular Ivan Petrovich Pavlov (1849-1936). In 1904 he won a Nobel Prize for his studies on the reflexes involved in digestion. But it was his discovery of conditioning, by which he made a considerable impact on the development not only psychology, but also AI [14].

The next person who made great contribution to the development of psychology and AI was Burrhus Frederic Skinner (1904-1990). He constructed a radical behaviourist theory in which behaviour is explained as the lawful result of environmental factors. Skinner is especially famous for the study of a form of learning known as operant conditioning [15].

The next three scientists who had a great impact on the development of behaviourism were Edward Lee Thorndike (1874-1949), Clark L. Hull (1884-1952) and Edward Chace Tolman (1886-1959). Their theories can be described as a 'subjective' behaviourism, because they moved away from the Skinner's radical behaviourism and in their explanation their refer to certain processes which take place within the organism.

Thorndike was particularly known for his extensive research into learning in animals and his attempt to develop a theoretical explanation for learning phenomena [16]. He initially described a form of learning know as trial-and-error learning or instrumental conditioning (Skinner used basically the same form of conditioning, but called it operant conditioning).

Hull is credited with developing the first systematic theory of learning known as the drive reduction theory [17]. According to his theory, it is drive and need that motivate to behave in a particular way.

According to Tolman [18], behaviour is largely regulated by cognitive factors such as the perception of signs and patterns in the environment, and the expectation of reward. Tolman can be regarded as a precursor of the social cognitive learning theory [19].

Social cognitive learning theory [20] agrees with other behaviouristically oriented theories in regarding behaviour as primarily learned and in focusing on the study of observable behaviour. However, there is a major difference because the social cognitive theory uses unobservable matters such as thoughts, expectations, and motivation in its explanation of behaviour. According to this school, the observational learning is the most important method of learning. Three psychologists, namely Julian Rotter, Albert Bandura and Walter Mischel are widely regarded as the most important figures in the development of social cognitive learning theory.

4. Learning

Learning can be defined as relatively permanent change in behaviour (both mental events and overt behaviours) that results from experience. Learning has taken place when a person or an animal has acquired knowledge of something that was previously unknown to him, or when he can do something he previously could not do.

The main types of learning that have been already identified and described can be classified on the basis of two criteria:

- The degree of understanding a learner must have of what is being learnt;
- The level of awareness on which learning takes place.

Two approaches to study learning have been used: association learning and cognitive learning.

In the association approach to learning, stimuli and responses are units on which the analysis of behavioural changes is based. The aim is to establish what the relationship is between a stimulus (S) and the human or animal organism's response (R) to it. There are two main types of association learning:

- Classical conditioning – I. P. Pavlov (1927);
- Operant conditioning – E.L. Thorndike (1913) [21] and B.F. Skinner (1969).

4.1 Classical Conditioning

Classical conditioning it is a kind of learning in which a neutral stimulus acquires the ability to produce a response that was originally produced by different stimulus.

The Russian physiologist Ivan Pavlov (1848-1936) is the father of classical conditioning. Pavlov first discovered that reflex of salivation and the secretion of gastric juices in a dog occur not only when food is placed in the dog's mouth, but also when the dog sees the food. He became interested in this phenomenon (Figure 1). In an experimental situation food was placed in a dog's mouth. Salivation occurred—salivation is a natural, reflexive and thus non-learned response to the stimulus (food). It occurred every time when food is given to the hungry dog. This response was named unconditioned response (UR). Pavlov then rang a bell close to the dog but as it was expected no salivation occurred. The sound of the bell is a neutral stimulus (NS). Later, Pavlov rang a bell before putting food in the dog's mouth. Salivation occurred. After a number of instances of hearing a bell paired with food, Pavlov again rang the bell, but he did not give food to the dog. Salivation occurred. In this situation salivation was elicited by the sound stimulus. Pavlov called this phenomenon a conditioned response (CR). The new S-R relationship (the relationship between the sound of the bell and salivation) is a consequence of the learned association between two stimuli (the bell and the food).

During classical conditioning, a dog not only learns to salivate to a tone but also simultaneously learns a number of other things that Pavlov identified as being a part of the classical conditioning procedure. The most important of them are: generalization, discrimination, extinction, and spontaneous recovery.

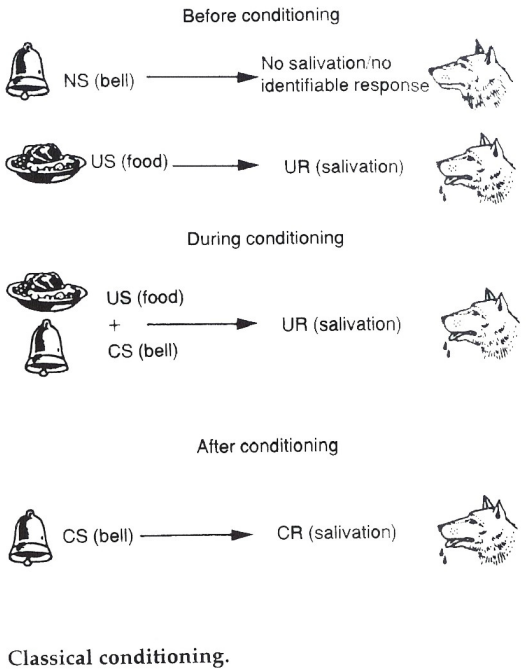


Figure 1. The process of Pavlovian classical conditioning. After [1]

Nowadays there are two explanations of classical conditioning: The traditional and modern.

Pavlov's traditional explanation is known as stimulus substitution. According to stimulus substitution, a bond or associations forms between the conditioned stimulus and unconditioned stimulus so that the conditioned stimulus eventually substituted for the unconditioned stimulus. Rescorla's modern explanation [23] is called stimulus information. According to the information theory of classical conditioning, an organism learns a relationship between two stimuli such that the occurrence of one stimulus predicts the occurrence of another.

4.2 Operant Conditioning (OC)

Operant Conditioning [9] it 'is a kind of learning in which the consequences that follow some behaviour increase or decrease the likelihood of that behaviour occurring in the future. In OC an organism (agent) acts or "operates" on the environment in order to change the likelihood of the response occurring again' (p.214).

The first steps in the development of operant conditioning are found in the work of Thorndike [21]. He formulated the law of effect, which stated that behaviours (goal-directed) followed by positive consequences are strengthened, while behaviours followed by negative consequences are weakened. Thorndike's ideas were further developed and expanded by Skinner. In a typical Skinner experiment a hungry pigeon is placed in a Skinner box. The pigeon walks around in the box, pecking here and there. Eventually the pigeon pecks against the lighted window and food falls into the bowl. By pecking against the lighted window the pigeon "operates" on its environment. Therefore this response is called an operant response. The food is the reward or reinforcer, which reinforces the appropriate response and increases the likelihood that pigeon will perform that behaviour in the future. After the reinforcer is presented a number of times,

immediately upon the appropriate pecking response, the probability of the pecking is greater than any other response. The procedure of behavioural shaping can be used in conditioning the pigeon to pick against the window. During shaping, the experimenter reinforces behaviours that lead up to or approximate the desired behaviour. The progress of operant conditioning can be divided into two phases (Figure 2 and 3).

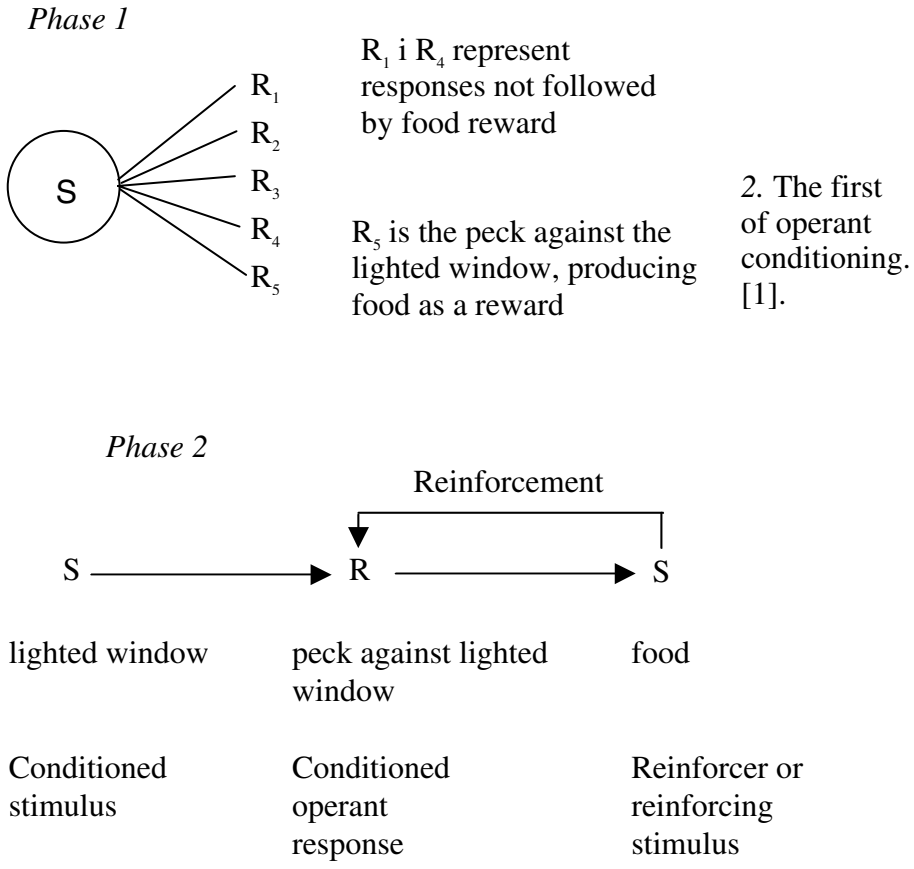


Figure 2. The first phrase of operant conditioning. After [1].

Figure 3. The second phrase of operant conditioning. After [1].

The first phrase corresponds to trial and error learning, in the sense that the pigeon produces the correct response by accident. The second phrase relates to the maintenance of the accidentally discovered correct response in accordance with the principle of reinforcement.

Operant conditioning focuses very sharply on the manipulability of behaviour. By manipulating the environmental conditions under which learning takes place, control can be exercised over the type and strength of behaviour that is learned. The essentials of operant conditioning can be summarized in five words—consequences are contingent on behaviour. There are two kinds of consequences—reinforcement and punishment. Reinforcement is a consequence that increases the likelihood of a behaviour occurring again and punishment is a consequence that decreases the likelihood of a behaviour occurring again. In addition, reinforcement can be either positive or negative. Positive reinforcement is a pleasant stimulus that increases the likelihood of a response occurring again. Negative reinforcement is the removal of an unpleasant stimulus, thereby increasing the likelihood of a response occurring again.

4.3 Cognitive Learning

Cognitive learning cannot be explained on the basis of reinforcing conditions. It is a kind of learning that involves mental processes, such as attention and memory, and may not involve any external rewards or require the person to perform any observable behaviours. Learning through thinking does not exclude the principles of association, however, it is regarded as a conscious act of thinking. There are two main kinds of cognitive learning:

- Sign/latent learning – E.C.Tolman (1932) [18],
- Observational learning – A.Bandura (1986) [23].

4.3.1 Sign/Latent Learning

According to Tolman, learning is attributed to the discovery of which response leads to what effect, and to a learned expectation that a certain stimulus will be followed by another stimulus. The stimuli are processed within the organism into an organized cognitive structure (cognitive map). The cognitive map is an organism's perceptual impressions of a learning situation. The performance of a correct response is a product of cognitive processes. Tolman also showed that organisms can learn in an absence of reinforcement—incidental learning. Bandura has developed Tolman's ideas.

4.3.2 Observational Learning

Bandura is a father of observational learning. It is a form of learning that develops through watching and does not require the observer to perform any observable behaviour or receive a reinforces. There are four components of observational learning: acquisition, retention, performance and reinforcement.

After describing the learning approaches the major differences between associative and cognitive learning are summarized now. Associative learning (behavioural approach) provides means of describing how a person or animal learns a series of correct or desired responses, it a kind of learning which demands little more than parrot-like repetitions under reinforcing conditions.

Cognitive learning explains learning with understanding and insight. Learning situation and material are perceptually organized by the learner, and then he formulates concepts and rules, next he recognizes the information so obtained into new and significant patters of information.

As a summary of learning, following Balkenius [24], we could present the main explanations of biological (animal) learning. According to those explanations the animal learns:

- Stimulus-response associations
- Stimulus-approach associations
- Place-approach associations
- Response chain
- Stimulus-approach chain
- Place-approach associations
- S-R-S' associations
- S-S' associations

However, we are not going to develop these aspects here.

5. Models of Behaviourists in AI

Robot learning in most cases is a kind an associative learning and differs widely, and ranges from the model of classical conditioning to reinforcement learning. Classical conditioning in robot learning is formed in the paradigm of unsupervised learning, for which learning rules are similar to *Hebb rules* or *Kohonen* ones. The Hebb rule [25] states that when a node (neuron) *i* repeatedly and persistently takes part in activating another node (neuron) *j*, then *i*-th neuron's efficiency in activating *j*-th neuron is increased. For example if a_i and a_j are the activations of the neuron *i* and *j*, respectively, and • is the learning rate, while w_{ij} is the connection weight (efficiency weight) between the neuron *i* and *j*, then the weight change Δw_{ij} is

$$\Delta w_{ij} = \bullet\, a_i\, a_j \qquad (1)$$

Hebbian learning has the advantage of being simple and based only on local communication between neurons: no central control is required. If a mobile robot, an agent, has been equipped with proximity and collision sensors as well as with a motor and wheels, the following so-called *distributed adaptive control architecture* can be interpreted in terms of classical conditioning (Figure 4).

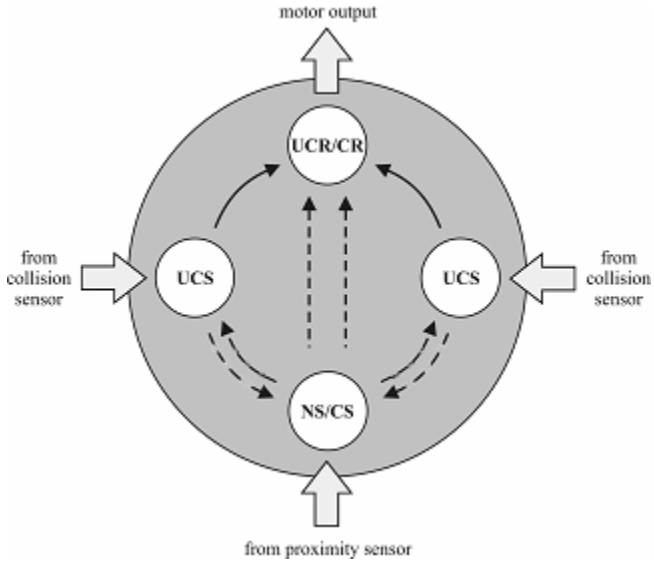


Figure 4. Distributed adaptive control architecture interpreted in terms of classical conditioning. after [10].

6. Distributed Adaptive Control

If a robot hits an obstacle as illustrated in Figure 4, activating a collision sensor, it backs up a little, turns and moves forward. Each sensor is connected to a node in the neural network: the collision sensors to nodes in the collision layer, the proximity sensors to nodes in the proximity layer. Moreover, the proximity layer is fully connected to the collision layer in one direction, while the collision layer is connected additionally to a motor output layer. The latter connections implement the basic reflexes, i.e. motor responses, called (in the Pavlovian approach) unconditioned response (UCR). The conditioned stimuli (CS) are the activations of the proximity sensors and the activations of the collision sensors model the unconditioned stimuli (UCS); they cause the robot to turn away from obstacle. We see that the UCS's are connected to the UCR's. (In Figure 4 collision sensors shown from both sides correspond to two sides of robots: the left side and the right one. The UCS - UCR connections are different for different sides since in one case turn in left, in the other turn in right, are executed to omit obstacle.) Note that before conditioning activation of proximity sensors (NS) did not cause any response (no action of motor devices). The conditioned response (CR), should, according to the Pavlovian approach, be very much like UCR; in the present (robot) situation it is not only similar to the UCR but in fact identical. (Notice that in Pavlov'e experiments the neutral CS (bell) was paired with a UCS (food) that reliably produced UCR (salivation), and after some trials the bell was sufficient to produce salivation (the CR).)

If at time *t* the robot hits an obstacle the corresponding node (neuron) in the collision layer is turned on, and simultaneously in several proximity nodes are activated. Then through Hebbian learning at the next time step *t+1* (cf. Eq. (1)) the corresponding connections between the proximity nodes (CS) and the active collision node (UCS) ($\Delta w_{ij=} w_{ij}(t+1) - w_{ij}(t_j)$ are strengthened. (In Figure 4 the learning stage is schematically presented by two curved arrows

pointing the bottom of the robot- NS/CS node.) This means that next time more activation from the CS nodes will be propagated to the UCS node. (Notice that when a collision appears the UCS nodes activate the UCR nodes in the motor output layer, and the robot backs up and turns to omit the obstacle.) If nodes in the collision layer (UCS) are binary threshold then after several hints due to the Hebbian rule of learning the activation originating from the CS layer becomes strong enough to raise the UCS node above threshold without collision initiating the activation the robot motor to avoid obstacles. (On Figure 4 this is schematically represented by two broken arrows pointing the top of the robot—the UCR/CR node.) When this happens the robot has learned to avoid obstacles through principle of classical conditioning [10].

Unsupervised learning paradigm known as a Kohonen map (or network) is used by a robot for location recognition, for example during which it measures the time between turn actions and hence it learns to recognize a particular location by building simple internal representations of its environment by a process of self-organization (without using explicit world model). In contrast to the previous distributed adaptive control, learning in the Kohonen algorithm is not incremental.

On the other hand operant learning in robot learning is formed in the paradigm of self-supervised learning. Here learning is based on reward (or punishment) resulting from behaviour. This is some-how similar to the operant conditioning of Thorndike and Skinner. Then two methods are distinguished, the first is that known in psychology as reinforcement learning, the other—as value-based learning. Value-based learning is learning modulated by a value system, which places some values on various types of sensory-actuator (motor) coordinations (i.e. value systems are activated only after an agent has performed behaviour. Both, however, methods employ principles of self-organization. The value system providing a kind of basic motivation for the agent guides the process of self-organization.

The third approach to learning is just teaching, and then we face with supervised learning or error-directed learning. For artificial, autonomous agent, such as neural networks that are models of human model behaviour, the delta rule, or more general, the error back-propagation rule are examples of learning rules. Often one says that supervised learning seems to resemble the way a mother teaches her child: The child can use the teaching signal from the mother to adjust his (her) responses. It is done by means of a supervised learning scheme in which the feedback from the mother has to be translated into error signals. However, this translation implies rather complex perceptual problems.

7. Conclusions

Following the point of view of the authors of [10] we can state that the complex intelligent behaviour can be performed by complete system (agent). According to them it should satisfied the following conditions:

- Complete system (agent) must possess the architecture:
  - with direct *coupling* of perception to action
  - with *dynamic interaction* with the environment
  - with intrinsic mechanisms to *cope* with resource limitations and incomplete knowledge
  - with *decentralized* processing

- Complete system has to be the *autonomous agent* (self-sufficient agent, equipped with the appropriate learning mechanism, with its own history, adaptive)
- Complete system has to be the situated *agent* (it acquires information about its environment only through its sensors and interacts with the world on its own)
- Complete system has to be *embodied* (it must interact with its environment, is continuously subjected to physical forces, to energy dissipation, to damage, to any influence in the environment)
- Complete system is behaviour based, not goal based
- Complete system includes sensors and effectors
- Sensory signals (stimuli) should be mapped (relatively) directly to effect or motors (responses)
- Complete system is equipped with a large number of parallel processes connected (only loosely) to one another

This leads to *embodied cognitive sciences* and to *embodied intelligence* introduced by Rodney Brooks [1991] and the *subsumption architecture.*

Since complete (i.e. intelligent) systems are behaviour based the behaviourists contributions are obvious.

Acknowledgement

The first author (W.K.) would like to thanks Mrs. and Mr. A. Adamkiewicz for the inspired discussions on psychology during his stay in their Dom na Ł kach in Izby, Beskid Niski Mountains, in the summer 2003. The help of Dr. Piotr Goł bek in collecting materials used in the preparation of the paper is highly acknowledged. The authors own a lot to Professor Rolf Pfeiefer from Stuttgart and his excellent *AI-lectures from Tokyo* he gave in the winter semester of 2003/2004 via a tele-conference system (and Internet) from Tokyo simultaneously to Zurich, Muenchen, Beijing and Warsaw.

References

1] Jordaan, W., Jordaan, J. (1994). Man in Context. Lexicon Publishers, Isando
[2] Spearman, C. (1904). General intelligence, objectively determined and measured. American Journal of Psychology, 15, 201-293
[3] Gregory, R, J. (1996). Psychological testing. History, principles and applications. Allyn & Bacon
[4] Terman, L.M. (1916). The measurement of intelligence. Boston: Houghton Mifflin.
[5] Pintner, R. (1921). Intelligence. In E.L. Thorndike (Ed.). Intelligence and Its measurement: A Symposium. Journal of Educational Psychology, 12, 123-147 and 195-216
[6] Thurstone, L.L. (1921). Intelligence. In E.L. Thorndike (Ed.). Intelligence and Its measurement: A Symposium. Journal of Educational Psychology, 12, 123-147 and 195-216
[7] Sternberg, R.J. (1986). Intelligence applied: Understanding and increasing your intellectual skills. S Diego, CA: Harcourt Brace Jovanovich
[8] Gardner, H. (1983). Frames of mind: The theory of multiple intelligence. New York: Basic Books
[9] Plotnik, R. (1993). Introduction to Psychology. Brooks/Cole Publishing Company, Pacific Grove, California
[10] Pfeifer, R., Scheier, Ch. (2001). Understanding Intelligence. MIT Press, Cambridge, Massachusetts, London, England
[11] Brooks, R.A. (1991). Intelligence without representation. Arificial Intelligence, 47, 139-160
[12] Arkin, R.C. (1998). Behavior-based robotics. Cambridge, MA: MIT Press
[13] Watson, J.B. (1913). Psychology as the behaviorist views it. Psychological Review, 20, 158-177
[14] Pavlov, I.P. (1927). Conditioned reflexes. London: Oxford
[15] Skinner, B.F. (1969). Contingencies of reinforcement. New York: Wiley
[16] Thorndike, E. L. (1911). Animal Intelligence. Hafner, Darien, Conn
[17] Hergenhahn, B.R. (1984). An introduction to theories of learning (2 ed.). Eglewood Cliffs, NJ: Prentice-Hall
[18] Tolman, E. C. (1932). Purposive Behavior in Animals and Men. Century, New York
[19] Brennan, J.F. (1982). History and systems of psychology. Eglewood Cliffs, NJ: Prentice-Hall
[20] Schults, D. (1990). Theories of personality (4ed.). Monterey, CAL: Brooks/Cole
[21] Thorndike, E.L. (1913). Educational psychology: The psychology of learning, vow.2. New York: Teachers College
[22] Rescorla, R.A. (1988). Pavlovian conditioning. American Psychologist, 43, 151-160
[23] Bandura, A. (1986). Social foundations of thought and action. Englewood Cliffs, NJ: Prentice Hall
[24] Balkenius, Ch. (1994). Biological learning and artificial intelligence. Lund Univerisity Cognitive Studies, 30, 1-19
[25] Hebb, E.O., (1949). The organization of behavior, New York: John Wiley and Sons

Chapter 10

The Emergence and Impact of Intelligent Machines

***

Raymond Kurzweil
Kurzweil Technologies
Wellesley Hills, Massachusetts, U.S.A.

The following issues are addressed in this essay.[1]

- **Models of Technology Trends**: A discussion of why nanotechnology and related advanced technologies are inevitable. The underlying technologies are deeply integrated into our society and are advancing on many diverse fronts.

- **The Economic Imperatives of the Law of Accelerating Returns**: The exponential advance of technology, including the accelerating miniaturization of technology, is driven by economic imperative, and, in turn, has a pervasive impact on the economy.

1. Models of Technology Trends

A diverse technology such as nanotechnology progresses on many fronts and is comprised of hundreds of small steps forward, each benign in itself. An examination of these trends shows that technology in which the key features are measured in a small number of nanometers is inevitable. I hereby provide some examples of my study of technology trends.

The motivation for this study came from my interest in inventing. As an inventor in the 1970s, I came to realize that my inventions needed to make sense in terms of the enabling technologies and market forces that would exist when the invention was introduced, which would represent a very different world than when it was conceived. I began to develop models of how distinct technologies—electronics, communications, computer processors, memory, magnetic storage, and the size of technology—developed and how these changes rippled through markets and ultimately our social institutions. I realized that most inventions fail not because they never work, but because their timing is wrong. Inventing is a lot like surfing, you have to anticipate and catch the wave at just the right moment.

[1] This chapter covers much of the material presented at a plenary address of the same title at the London conference on *Intelligent Motion and Interaction within Virtual Environments*. More of Kurzweil's writing may be found on the web at: http://www.kurzweilai.net/meme/frame.html?m=10.

In the 1980s, my interest in technology trends and implications took on a life of its own, and I began to use my models of technology trends to project and anticipate the technologies of future times, such as the year 2000, 2010, 2020, and beyond. This enabled me to invent with the capabilities of the future. In the late 1980s, I wrote my first book, *The Age of Intelligent Machines*, which ended with the specter of machine intelligence becoming indistinguishable from its human progenitors. This book included hundreds of predictions about the 1990s and early 2000 years, and my track record of prediction has held up well.

During the 1990s I gathered empirical data on the apparent acceleration of all information-related technologies and sought to refine the mathematical models underlying these observations. In *The Age of Spiritual Machines* (ASM), which I wrote in 1998, I introduced refined models of technology, and a theory I called "the law of accelerating returns," which explained why technology evolves in an exponential fashion.

1.1 The Intuitive Linear View versus the Historical Exponential View

The future is widely misunderstood. Our forebears expected the future to be pretty much like their present, which had been pretty much like their past. Although exponential trends did exist a thousand years ago, they were at that very early stage where an exponential trend is so flat and so slow that it looks like no trend at all. So their lack of expectations was largely fulfilled. Today, in accordance with the common wisdom, everyone expects continuous technological progress and the social repercussions that follow. But the future will nonetheless be far more surprising than most observers realize because few have truly internalized the implications of the fact that the rate of change itself is accelerating.

Most long-range forecasts of technical feasibility in future time periods dramatically underestimate the power of future developments because they are based on what I call the "intuitive linear" view of history rather than the "historical exponential view." To express this another way, it is not the case that we will experience a hundred years of progress in the twenty-first century; rather we will witness on the order of twenty thousand years of progress (at *today's* rate of progress, that is).

When people think of a future period, they intuitively assume that the current rate of progress will continue for future periods. Even for those who have been around long enough to experience how the pace increases over time, an unexamined intuition nonetheless provides the impression that progress changes at the rate that we have experienced recently. From the mathematician's perspective, a primary reason for this is that an exponential curve approximates a straight line when viewed for a brief duration. It is typical, therefore, that even sophisticated commentators, when considering the future, extrapolate the current pace of change over the next 10 years or 100 years to determine their expectations. This is why I call this way of looking at the future the "intuitive linear" view.

But a serious assessment of the history of technology shows that technological change is exponential. In exponential growth, we find that a key measurement such as computational power is multiplied by a constant factor for each unit of time (e.g., doubling every year) rather than just being added to incrementally. Exponential growth is a feature of any evolutionary

process, of which technology is a primary example. One can examine the data in different ways, on different time scales, and for a wide variety of technologies ranging from electronic to biological, as well as social implications ranging from the size of the economy to human life span, and the acceleration of progress and growth applies. Indeed, we find not just simple exponential growth, but "double" exponential growth, meaning that the rate of exponential growth is itself growing exponentially. These observations do not rely merely on an assumption of the continuation of Moore's law (i.e., the exponential shrinking of transistor sizes on an integrated circuit), but is based on a rich model of diverse technological processes. What it clearly shows is that technology, particularly the pace of technological change, advances (at least) exponentially, not linearly, and has been doing so since the advent of technology, indeed since the advent of evolution on Earth.

Many scientists and engineers have what my colleague Lucas Hendrich calls "engineer's pessimism." Often an engineer or scientist who is so immersed in the difficulties and intricate details of a contemporary challenge fails to appreciate the ultimate long-term implications of their own work, and, in particular, the larger field of work that they operate in. Consider the biochemists in 1985 who were skeptical of the announcement of the goal of transcribing the entire genome in a mere 15 years. These scientists had just spent an entire year transcribing a mere one ten-thousandth of the genome, so even with reasonable anticipated advances, it seemed to them like it would be hundreds of years, if not longer, before the entire genome could be sequenced. Or consider the skepticism expressed in the mid 1980s that the Internet would ever be a significant phenomenon, given that it included only tens of thousands of nodes. The fact that the number of nodes was doubling every year and there were, therefore, likely to be tens of millions of nodes ten years later was not appreciated by those who struggled with "state of the art" technology in 1985, which permitted adding only a few thousand nodes throughout the world in a year.

I emphasize this point because it is the most important failure that would-be prognosticators make in considering future trends. The vast majority of technology forecasts and forecasters ignore altogether this "historical exponential view" of technological progress. Indeed, almost everyone I meet has a linear view of the future. That is why people tend to overestimate what can be achieved in the short term (because we tend to leave out necessary details), but underestimate what can be achieved in the long term (because the exponential growth is ignored).

1.2 The Law of Accelerating Returns

The ongoing acceleration of technology is the implication and inevitable result of what I call the "law of accelerating returns," which describes the acceleration of the pace and the exponential growth of the products of an evolutionary process. This includes technology, particularly information-bearing technologies, such as computation. More specifically, the law of accelerating returns states the following:

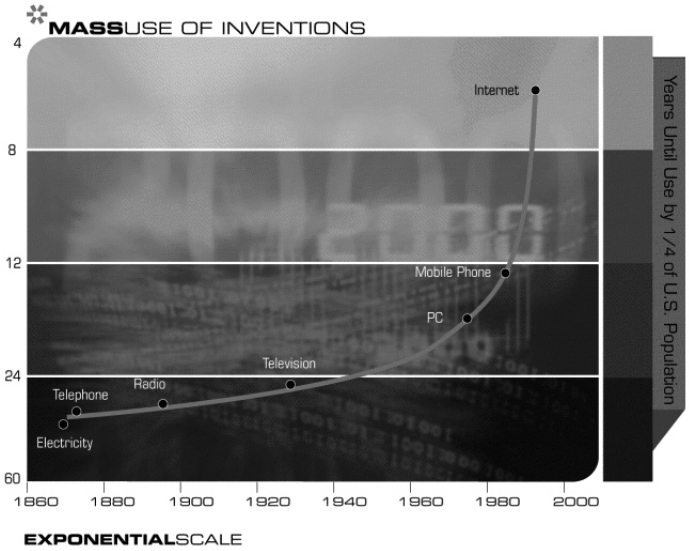
- Evolution applies positive feedback in that the more capable methods resulting from one stage of evolutionary progress are used to create the next stage. As a result, the rate of progress of an evolutionary process increases exponentially over time. Over time, the "order" of the information embedded in the evolutionary process (i.e., the measure of how well the information fits a purpose, which in evolution is survival) increases.

- A correlate of the above observation is that the "returns" of an evolutionary process (e.g., the speed, cost-effectiveness, or overall "power" of a process) increase exponentially over time.
- In another positive feedback loop, as a particular evolutionary process (e.g., computation) becomes more effective (e.g., cost effective), greater resources are deployed towards the further progress of that process. This results in a second level of exponential growth (i.e., the rate of exponential growth itself grows exponentially).
- Biological evolution is one such evolutionary process.
- Technological evolution is another such evolutionary process. Indeed, the emergence of the first technology-creating species resulted in the new evolutionary process of technology. Therefore, technological evolution is an outgrowth of – and a continuation of – biological evolution.
- A specific paradigm (a method or approach to solving a problem, e.g., shrinking transistors on an integrated circuit as an approach to making more powerful computers) provides exponential growth until the method exhausts its potential. When this happens, a paradigm shift (a fundamental change in the approach) occurs, which enables exponential growth to continue.
- Each paradigm follows an "S-curve," which consists of slow growth (the early phase of exponential growth), followed by rapid growth (the late, explosive phase of exponential growth), followed by a leveling off as the particular paradigm matures.
- During this third or maturing phase in the life cycle of a paradigm, pressure builds for the next paradigm shift.
- When the paradigm shift occurs, the process begins a new S-curve.
- Thus the acceleration of the overall evolutionary process proceeds as a sequence of S-curves, and the overall exponential growth consists of this cascade of S-curves.
- The resources underlying the exponential growth of an evolutionary process are relatively unbounded.
- One resource is the (ever-growing) order of the evolutionary process itself. Each stage of evolution provides more powerful tools for the next. In biological evolution, the advent of DNA allowed more powerful and faster evolutionary "experiments." Later, setting the "designs" of animal body plans during the Cambrian explosion allowed rapid evolutionary development of other body organs, such as the brain. Or to take a more recent example, the advent of computer-assisted design tools allows rapid development of the next generation of computers.
- The other required resource is the "chaos" of the environment in which the evolutionary process takes place and which provides the options for further diversity. In biological evolution, diversity enters the process in the form of mutations and ever- changing environmental conditions, including cosmological disasters (e.g., asteroids hitting the Earth). In technological evolution, human ingenuity combined with ever-changing market conditions keep the process of innovation going.

If we apply these principles at the highest level of evolution on Earth, the first step, the creation of cells, introduced the paradigm of biology. The subsequent emergence of DNA provided a digital method to record the results of evolutionary experiments. Then, the evolution of a species that combined rational thought with an opposable appendage (the thumb) caused a fundamental paradigm shift from biology to technology. The upcoming primary paradigm shift will be from biological thinking to a hybrid combining biological and nonbiological thinking. This hybrid will include "biologically inspired" processes resulting from the reverse engineering of biological brains.

If we examine the timing of these steps, we see that the process has continuously accelerated. The evolution of life forms required billions of years for the first steps (e.g., primitive cells); later on progress accelerated. During the Cambrian explosion, major paradigm shifts took only tens of millions of years. Later on, Humanoids developed over a period of millions of years, and Homo sapiens over a period of only hundreds of thousands of years.

With the advent of a technology-creating species, the exponential pace became too fast for evolution through DNA-guided protein synthesis and moved on to human-created technology. Technology goes beyond mere tool making; it is a process of creating ever more powerful technology using the tools from the previous round of innovation, and is, thereby, an evolutionary process. The first technological steps—sharp edges, fire, the wheel—took tens of thousands of years. For people living in this era, there was little noticeable technological change in even a thousand years. By 1000 AD, progress was much faster and a paradigm shift required only a century or two. In the nineteenth century, we saw more technological change than in the nine centuries preceding it. Then in the first twenty years of the twentieth century, we saw more advancement than in all of the nineteenth century. Now, paradigm shifts occur in only a few years time. The World Wide Web did not exist in anything like its present form just a few years ago; it didn't exist at all a decade ago.



The paradigm shift rate (i.e., the overall rate of technical progress) is currently doubling (approximately) every decade; that is, paradigm shift times are halving every decade (and the rate of acceleration is itself growing exponentially). So, the technological progress in the twenty-first century will be equivalent to what would require (in the linear view) on the order of 200 centuries. In contrast, the twentieth century saw only about 20 years of progress (again at today's rate of progress) since we have been speeding up to current rates. So the twenty-first century will see about a thousand times greater technological change than its predecessor.

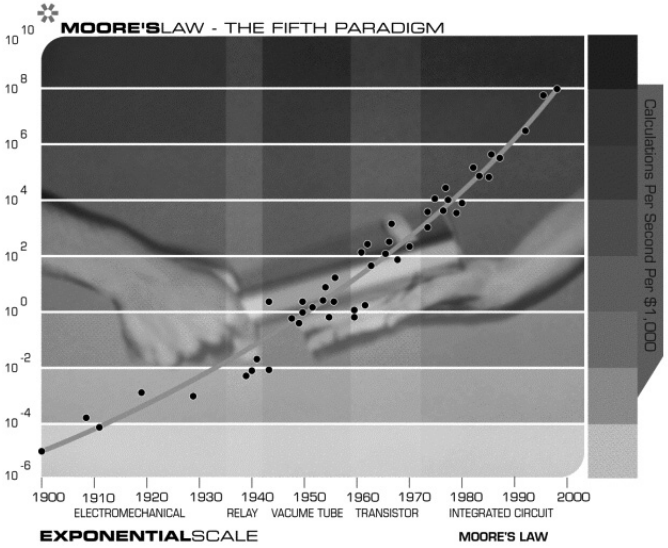
1.3 Moore's Law and Beyond

There is a wide range of technologies that are subject to the law of accelerating returns. The exponential trend that has gained the greatest public recognition has become known as "Moore's Law." Gordon Moore, one of the inventors of integrated circuits, and then Chairman of Intel, noted in the mid-1970s that we could squeeze twice as many transistors on an integrated circuit every 24 months. Given that the electrons have less distance to travel, the circuits also run twice as fast, providing an overall quadrupling of computational power.

However, the exponential growth of computing is much broader than Moore's Law.

If we plot the speed (in instructions per second) per $1000 (in constant dollars) of 49 famous calculators and computers spanning the entire twentieth century, we note that there were four completely different paradigms that provided exponential growth in the price-performance of computing before the integrated circuits were invented. Therefore, Moore's Law was not the first, but the fifth paradigm to exponentially grow the power of computation. And it won't be the last. When Moore's Law reaches the end of its S-Curve, now expected before 2020, the exponential growth will continue with three-dimensional molecular computing, a prime example of the application of nanotechnology, which will constitute the sixth paradigm.

When I suggested in my book *The Age of Spiritual Machines,* published in 1999, that three-dimensional molecular computing, particularly an approach based on using carbon nanotubes, would become the dominant computing hardware technology in the teen years of this century, that was considered a radical notion. There has been so much progress in the past four years, with literally dozens of major milestones having been achieved, that this expectation is now a mainstream view.



Moore's Law was not the first, but the fifth paradigm to provide exponential growth of computing. Each time one paradigm runs out of steam, another picks up the pace.

The exponential growth of computing is a marvelous quantitative example of the exponentially growing returns from an evolutionary process. We can express the exponential growth of computing in terms of an accelerating pace: it took 90 years to achieve the first MIPS (million instructions per second) per thousand dollars; now we add one MIPS per thousand dollars every day.

Moore's Law narrowly refers to the number of transistors on an integrated circuit of fixed size, and sometimes has been expressed even more narrowly in terms of transistor feature size. But rather than feature size (which is only one contributing factor), or even number of transistors, I think the most appropriate measure to track is computational speed per unit cost. This takes into account many levels of "cleverness" (i.e., innovation, which is to say, technological evolution). In addition to all of the innovation in integrated circuits, there are multiple layers of innovation in computer design, e.g., pipelining, parallel processing, instruction look-ahead, instruction and memory caching, and many others.

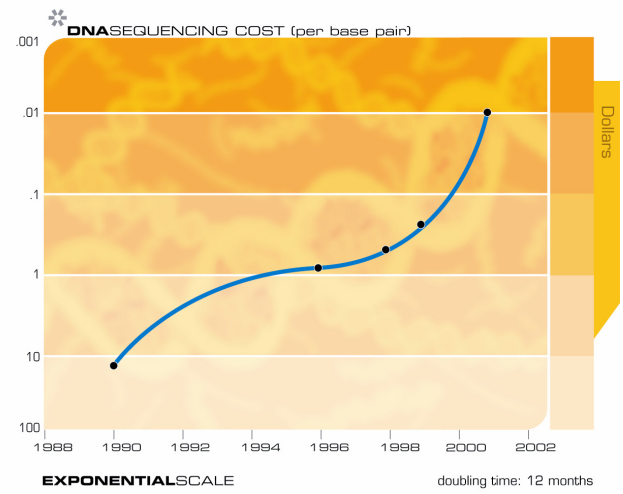
The human brain uses a very inefficient electrochemical digital-controlled analog computational process. The bulk of the calculations are done in the interneuronal connections at a speed of only about 200 calculations per second (in each connection), which is about ten million times slower than contemporary electronic circuits. But the brain gains its prodigious powers from its extremely parallel organization *in three dimensions.* There are many technologies in the wings that build circuitry in three dimensions. Nanotubes, an example of nanotechnology, which is already working in laboratories, build circuits from pentagonal arrays of carbon atoms. One cubic inch of nanotube circuitry would be a million times more powerful than the human brain. There are more than enough new computing technologies now being researched, including three-dimensional silicon chips, optical and silicon spin computing, crystalline computing, DNA computing, and quantum computing, to keep the law of accelerating returns as applied to computation going for a long time.

As I discussed previously, it is important to distinguish between the "S" curve (an "S" stretched to the right, comprising very slow, virtually unnoticeable growth—followed by very rapid growth— followed by a flattening out as the process approaches an asymptote) that is characteristic of any specific technological paradigm and the continuing exponential growth that is characteristic of the ongoing evolutionary process of technology. Specific paradigms, such as Moore's Law, do ultimately reach levels at which exponential growth is no longer feasible. That is why Moore's Law is an S curve. But the growth of computation is an ongoing exponential (at least until we "saturate" the Universe with the intelligence of our human-machine civilization, but that will not be a limit in this coming century). In accordance with the law of accelerating returns, paradigm shift, also called innovation, turns the S curve of any specific paradigm into a continuing exponential. A new paradigm (e.g., three-dimensional circuits) takes over when the old paradigm approaches its natural limit, which has already happened at least four times in the history of computation. This difference also distinguishes the tool making of non-human species, in which the mastery of a tool-making (or using) skill by each animal is characterized by an abruptly ending S shaped learning curve, versus human-created technology, which has followed an exponential pattern of growth and acceleration since its inception.

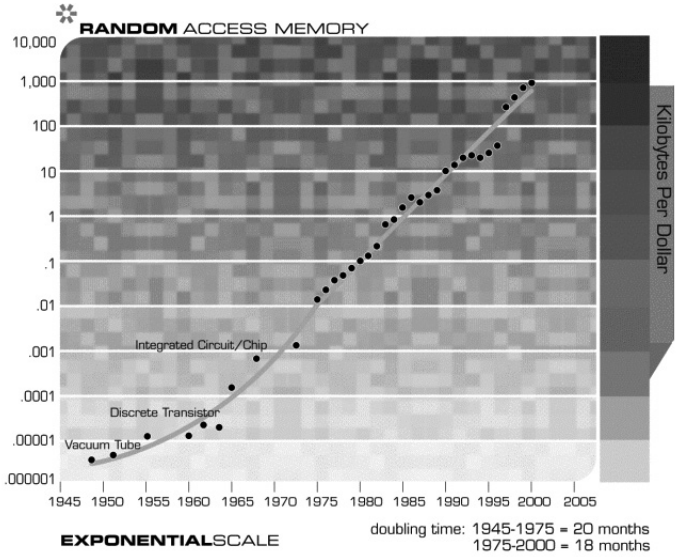
1.4 DNA Sequencing, Memory, Communications, the Internet, and Miniaturization

This "law of accelerating returns" applies to all of technology, indeed to any true evolutionary process, and can be measured with remarkable precision in information-based technologies. There are a great many examples of the exponential growth implied by the law of accelerating returns in technologies, as varied as DNA sequencing, communication speeds, brain scanning, electronics of all kinds, and even in the rapidly shrinking size of technology, which is directly relevant to the discussion at this hearing. The future nanotechnology age results not from the exponential explosion of computation alone, but rather from the interplay and myriad synergies that will result from manifold intertwined technological revolutions. Also, keep in mind that every point on the exponential growth curves underlying these panoply of technologies (see the graphs below) represents an intense human drama of innovation and competition. It is remarkable therefore that these chaotic processes result in such smooth and predictable exponential trends.

As I noted above, when the human genome scan started fourteen years ago, critics pointed out that given the speed with which the genome could then be scanned, it would take thousands of years to finish the project. Yet the fifteen year project was nonetheless completed slightly ahead of schedule.

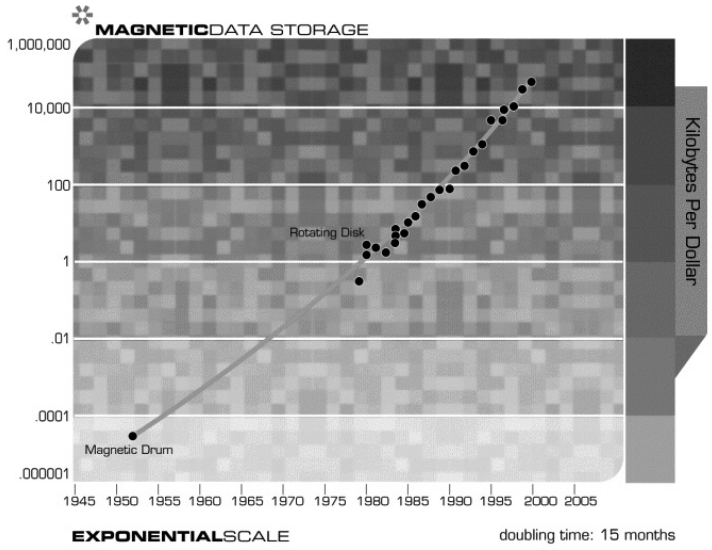


Of course, we expect to see exponential growth in electronic memories such as RAM.

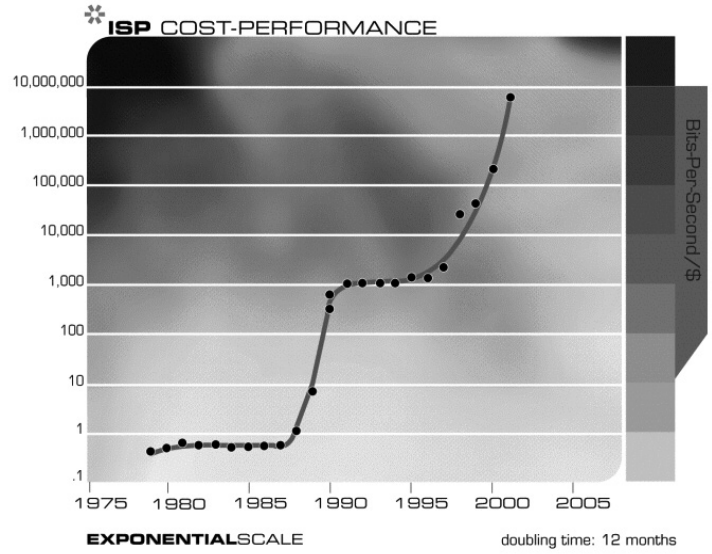


Notice How Exponential Growth Continued through Paradigm Shifts from Vacuum Tubes to Discrete Transistors to Integrated Circuits

However, growth in magnetic memory is not primarily a matter of Moore's law, but includes advances in mechanical and electromagnetic systems.



Exponential growth in communications technology has been even more explosive than in computation and is no less significant in its implications. Again, this progression involves far more than just shrinking transistors on an integrated circuit, but includes accelerating advances in fiber optics, optical switching, electromagnetic technologies, and others.

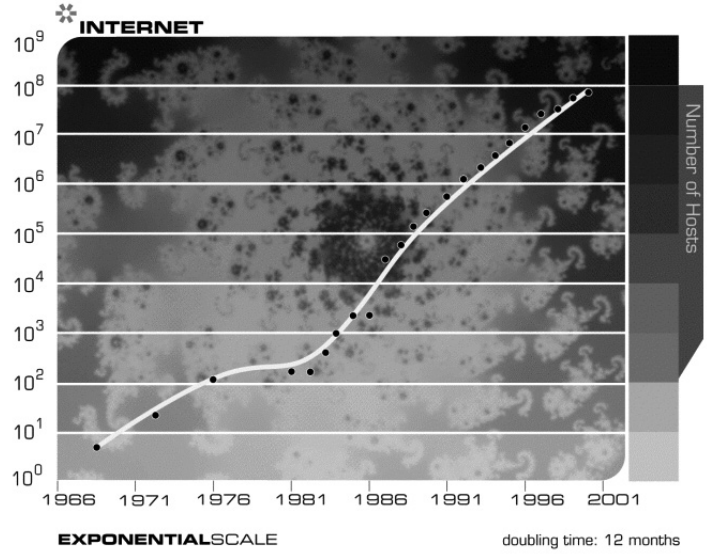


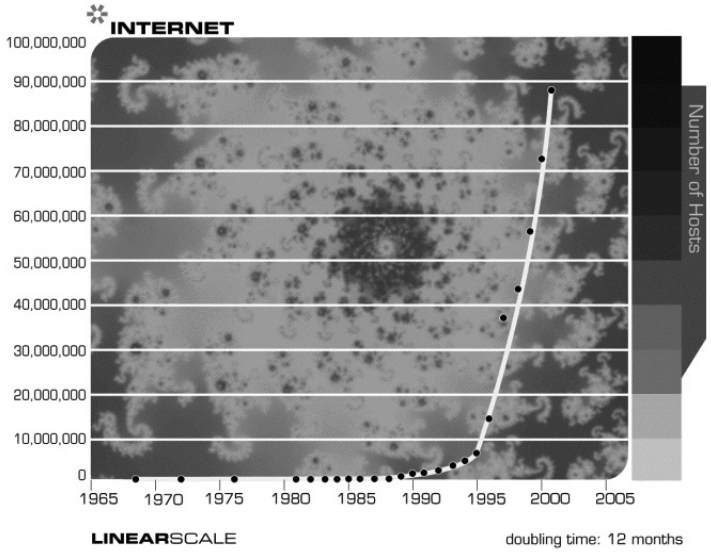
Notice Cascade of "S" Curves.

Note that in the above chart we can actually see the progression of "S" curves: the acceleration fostered by a new paradigm, followed by a leveling off as the paradigm runs out of steam, followed by renewed acceleration through paradigm shift.

The following two charts show the overall growth of the Internet based on the number of hosts (server computers). These two charts plot the same data, but one is on an exponential axis and the other is linear. As I pointed out earlier, whereas technology progresses in the exponential domain, we experience it in the linear domain. So from the perspective of most observers,

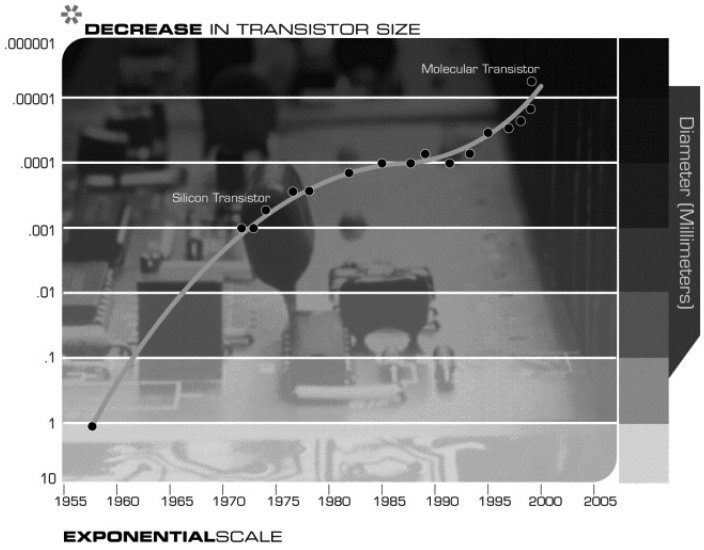
nothing was happening until the mid 1990s when seemingly out of nowhere, the World Wide Web and email exploded into view. But the emergence of the Internet into a worldwide phenomenon was readily predictable much earlier by examining the exponential trend data.

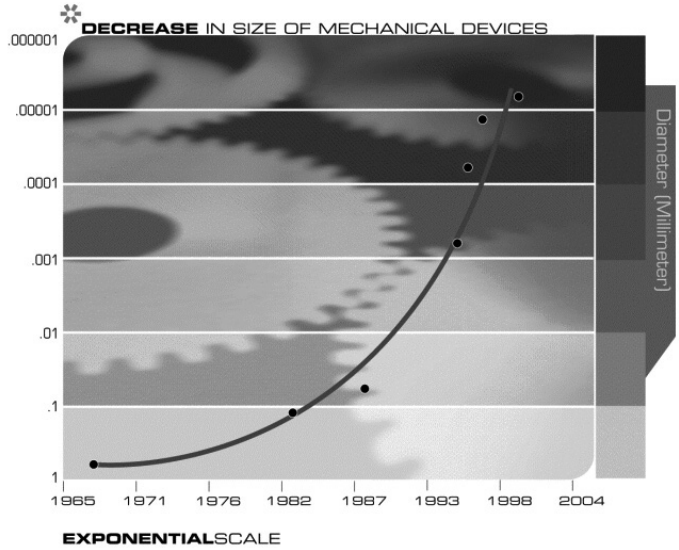




Notice how the explosion of the Internet appears to be a surprise from the Linear Chart, but was perfectly predictable from the Exponential Chart.

The most relevant trend to this hearing, and one that will have profound implications for the twenty-first century is the pervasive trend towards making things smaller, i.e., miniaturization. The salient implementation sizes of a broad range of technologies, both electronic and mechanical, are shrinking, also at a double-exponential rate. At present, we are shrinking technology by a factor of approximately 5.6 per linear dimension per decade.





2. The Economic Imperatives of the Law of Accelerating Returns

It is the economic imperative of a competitive marketplace that is driving technology forward and fueling the law of accelerating returns. In turn, the law of accelerating returns is transforming economic relationships.

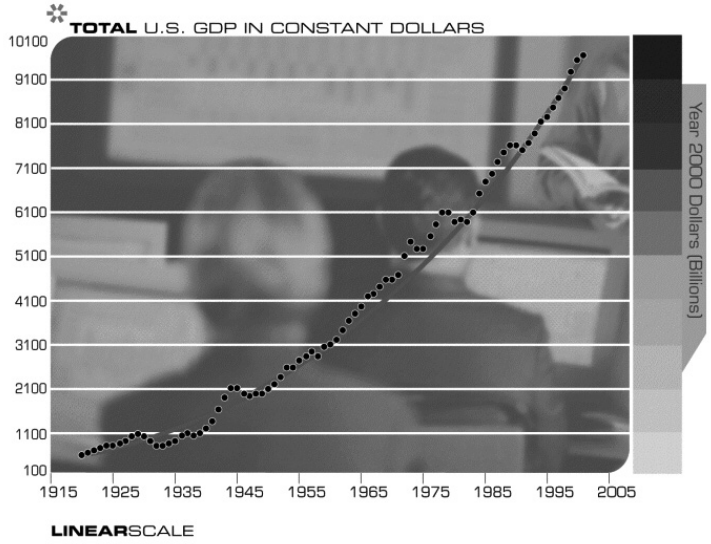
The primary force driving technology is economic imperative. We are moving towards nanoscale machines, as well as more intelligent machines, as the result of a myriad of small advances, each with their own particular economic justification.

To use one small example of many from my own experience at one of my companies (Kurzweil Applied Intelligence), whenever we came up with a slightly more intelligent version of speech recognition, the new version invariably had greater value than the earlier generation and, as a

result, sales increased. It is interesting to note that in the example of speech recognition software, the three primary surviving competitors stayed very close to each other in the intelligence of their software. A few other companies that failed to do so (e.g., Speech Systems) went out of business. At any point in time, we would be able to sell the version prior to the latest version for perhaps a quarter of the price of the current version. As for versions of our technology that were two generations old, we couldn't even give those away.

There is a vital economic imperative to create smaller and more intelligent technology. Machines that can more precisely carry out their missions have enormous value. That is why they are being built. There are tens of thousands of projects that are advancing the various aspects of the law of accelerating returns in diverse incremental ways. Regardless of near-term business cycles, the support for "high tech" in the business community, and in particular for software advancement, has grown enormously. When I started my optical character recognition (OCR) and speech synthesis company (Kurzweil Computer Products, Inc.) in 1974, high-tech venture deals totaled approximately $10 million. Even during today's high tech recession, the figure is 100 times greater. We would have to repeal capitalism and every visage of economic competition to stop this progression.

The economy (viewed either in total or per capita) has been growing exponentially throughout this century:



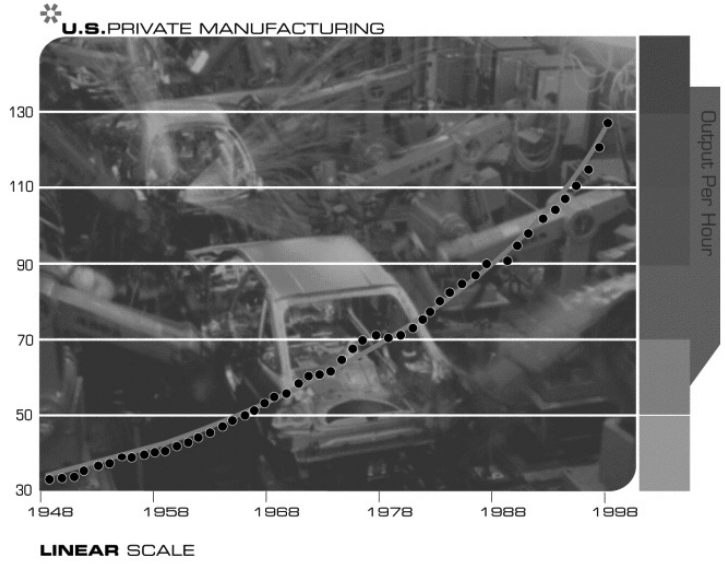
Note that the underlying exponential growth in the economy is a far more powerful force than periodic recessions. Even the "Great Depression" represents only a minor blip compared to the underlying pattern of growth. Most importantly, recessions, including the depression, represent only temporary deviations from the underlying curve. In each case, the economy ends up exactly where it would have been had the recession/depression never occurred.

Productivity (economic output per worker) has also been growing exponentially. Even these statistics are greatly understated because they do not fully reflect significant improvements in the quality and features of products and services. It is not the case that "a car is a car;" there have been significant improvements in safety, reliability, and features. Certainly, $1000 of

computation today is immeasurably more powerful than $1000 of computation ten years ago (by a factor of more than1000). There are a myriad of such examples. Pharmaceutical drugs are increasingly effective. Products ordered in five minutes on the web and delivered to your door are worth more than products that you have to fetch yourself. Clothes custom-manufactured for your unique body scan are worth more than clothes you happen to find left on a store rack. These sorts of improvements are true for most product categories, and none of them are reflected in the productivity statistics.

The statistical methods underlying the productivity measurements tend to factor out gains by essentially concluding that we still only get one dollar of products and services for a dollar despite the fact that we get much more for a dollar (e.g., compare a $1,000 computer today to one ten years ago). University of Chicago Professor Pete Klenow and University of Rochester Professor Mark Bils estimate that the value of existing goods has been increasing at 1.5% per year for the past 20 years because of qualitative improvements. This still does not account for the introduction of entirely new products and product categories (e.g., cell phones, pagers, pocket computers). The Bureau of Labor Statistics, which is responsible for the inflation statistics, uses a model that incorporates an estimate of quality growth at only 0.5% per year, reflecting a systematic underestimate of quality improvement and a resulting overestimate of inflation by at least 1 percent per year.

Despite these weaknesses in the productivity statistical methods, the gains in productivity are now reaching the steep part of the exponential curve. Labor productivity grew at 1.6% per year until 1994, then rose at 2.4% per year, and is now growing even more rapidly. In the quarter ending July 30, 2000, labor productivity grew at 5.3%. Manufacturing productivity grew at 4.4% annually from 1995 to 1999, durables manufacturing at 6.5% per year.



The 1990s have seen the most powerful deflationary forces in history. This is why we are not seeing inflation. Yes, it's true that low unemployment, high asset values, economic growth, and other such factors are inflationary, but these factors are offset by the double-exponential trends in the price-performance of all information-based technologies: computation, memory,

communications, biotechnology, miniaturization, and even the overall rate of technical progress. These technologies deeply affect all industries. We are also undergoing massive disintermediation in the channels of distribution through the Web and other new communication technologies, as well as escalating efficiencies in operations and administration.

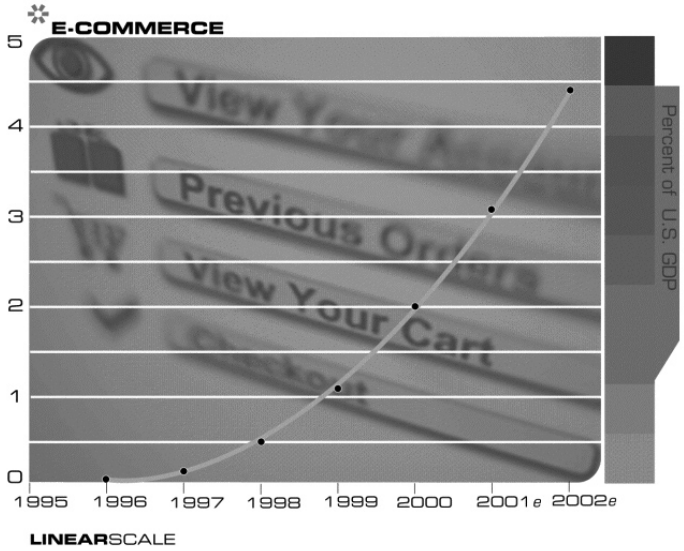
All of the technology trend charts above represent massive deflation. There are many examples of the impact of these escalating efficiencies. BP Amoco's cost for finding oil is now less than $1 per barrel, down from nearly $10 in 1991. Processing an Internet transaction costs a bank one penny, compared to over $1 using a teller ten years ago. A Roland Berger/Deutsche Bank study estimates a cost savings of $1200 per North American car over the next five years. A more optimistic Morgan Stanley study estimates that Internet-based procurement will save Ford, GM, and DaimlerChrysler about $2700 per vehicle.

It is important to point out that a key implication of nanotechnology is that it will bring the economics of software to hardware, i.e., to physical products. Software prices are deflating even more quickly than hardware.

Software Price-Performance Has Also Improved at an Exponential Rate:

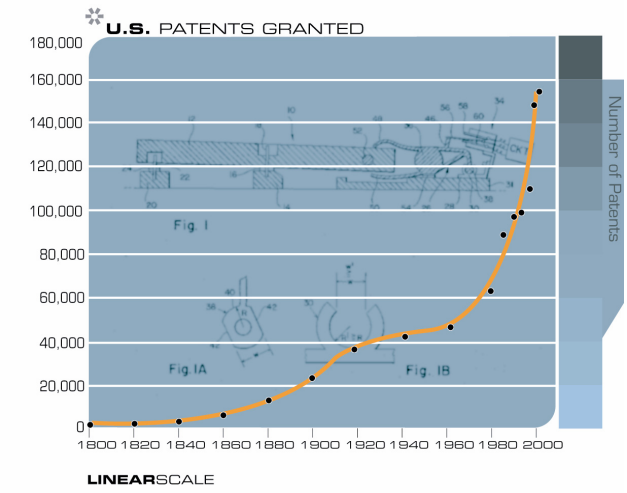
Automatic Speech Recognition Software

	1985	1995	2000
Price	$5,000	$500	$50
Vocabulary size (# words)	1,000	10,000	100,000
Continuous speech?	No	No	Yes
User training required (minutes)	180	60	5
Accuracy	Poor	Fair	Good



Current economic policy is based on outdated models that include energy prices, commodity prices, and capital investment in plant and equipment as key driving factors, but do not adequately model the size of technology, bandwidth, MIPs, megabytes, intellectual property, knowledge, and other increasingly vital (and increasingly increasing) constituents that are driving the economy.

Another indication of the law of accelerating returns in the exponential growth of human knowledge, including intellectual property. If we look at the development of intellectual property within the nanotechnology field, we see even more rapid growth.

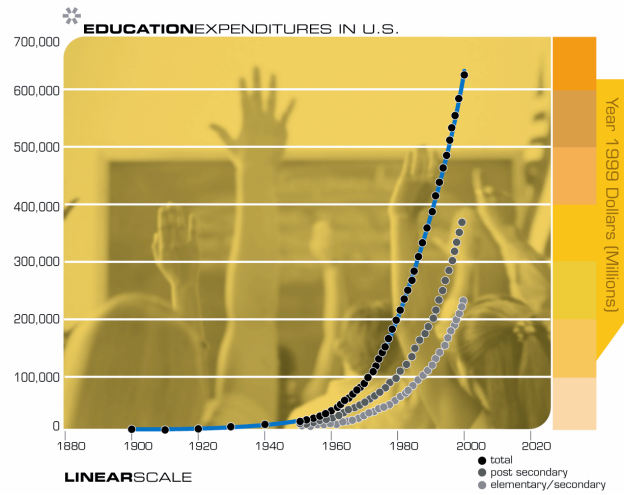


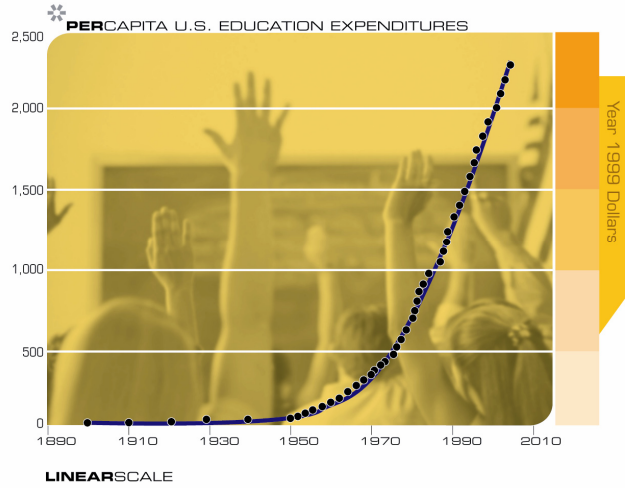
None of this means that cycles of recession will disappear immediately. Indeed there is a current economic slowdown and a technology-sector recession. The economy still has some of the underlying dynamics that historically have caused cycles of recession, specifically excessive commitments such as over-investment, excessive capital intensive projects and the overstocking of inventories. However, the rapid dissemination of information, sophisticated forms of online procurement, and increasingly transparent markets in all industries have diminished the impact of this cycle. So “recessions” are likely to have less direct impact on our standard of living. The underlying long-term growth rate will continue at a double exponential rate.

Moreover, innovation and the rate of paradigm shift are not noticeably affected by the minor deviations caused by economic cycles. All of the technologies exhibiting exponential growth shown in the above charts are continuing without losing a beat through this economic slowdown.

The overall growth of the economy reflects completely new forms and layers of wealth and value that did not previously exist, or least that did not previously constitute a significant portion of the economy (but do now): new forms of nanoparticle-based materials, genetic information, intellectual property, communication portals, web sites, bandwidth, software, data bases, and many other new technology-based categories.

Another implication of the law of accelerating returns is exponential growth in education and learning. Over the past 120 years, we have increased our investment in K-12 education (per student and in constant dollars) by a factor of ten. We have a one hundred fold increase in the number of college students. Automation started by amplifying the power of our muscles, and in recent times has been amplifying the power of our minds. Thus, for the past two centuries, automation has been eliminating jobs at the bottom of the skill ladder while creating new (and better paying) jobs at the top of the skill ladder. So the ladder has been moving up, and thus we have been exponentially increasing investments in education at all levels.





3. The Deeply Intertwined Promise and Peril of Nanotechnology and Related Advanced Technologies

Technology has always been a double-edged sword, bringing us longer and healthier life spans, freedom from physical and mental drudgery, and many new creative possibilities on the one hand, while introducing new and salient dangers on the other. Technology empowers both our creative and destructive natures. Stalin's tanks and Hitler's trains used technology. We still live

today with sufficient nuclear weapons (not all of which appear to be well accounted for) to end all mammalian life on the planet. Bioengineering is in the early stages of enormous strides in reversing disease and aging processes. However, the means and knowledge will soon exist in a routine college bioengineering lab (and already exists in more sophisticated labs) to create unfriendly pathogens more dangerous than nuclear weapons. As technology accelerates towards the full realization of biotechnology, nanotechnology and "strong" AI (artificial intelligence at human levels and beyond), we will see the same intertwined potentials: a feast of creativity resulting from human intelligence expanded many-fold combined with many grave new dangers.

Consider unrestrained nanobot replication. Nanobot technology requires billions or trillions of such intelligent devices to be useful. The most cost-effective way to scale up to such levels is through self-replication, essentially the same approach used in the biological world. And in the same way that biological self-replication gone awry (i.e., cancer) results in biological destruction, a defect in the mechanism curtailing nanobot self-replication would endanger all physical entities, biological or otherwise. I address below steps we can take to address this grave risk, but we cannot have complete assurance in any strategy that we devise today.

Other primary concerns include "who is controlling the nanobots?" and "who are the nanobots talking to?" Organizations (e.g., governments, extremist groups) or just a clever individual could put trillions of undetectable nanobots in the water or food supply of an individual or of an entire population. These "spy" nanobots could then monitor, influence, and even control our thoughts and actions. In addition to introducing physical spy nanobots, existing nanobots could be influenced through software viruses and other software "hacking" techniques. When there is software running in our brains, issues of privacy and security will take on a new urgency.

My own expectation is that the creative and constructive applications of this technology will dominate, as I believe they do today. However, I believe we need to invest more heavily in developing specific defensive technologies. As I address further below, we are at this stage today for biotechnology, and will reach the stage where we need to directly implement defensive technologies for nanotechnology during the late teen years of this century.

If we imagine describing the dangers that exist today to people who lived a couple of hundred years ago, they would think it mad to take such risks. On the other hand, how many people in the year 2000 would really want to go back to the short, brutish, disease-filled, poverty-stricken, disaster-prone lives that 99 percent of the human race struggled through a couple of centuries ago? We may romanticize the past, but up until fairly recently, most of humanity lived extremely fragile lives where one all-too-common misfortune could spell disaster. Substantial portions of our species still live in this precarious way, which is at least one reason to continue technological progress and the economic enhancement that accompanies it.

People often go through three stages in examining the impact of future technology: awe and wonderment at its potential to overcome age old problems; then a sense of dread at a new set of grave dangers that accompany these new technologies; followed, finally and hopefully, by the realization that the only viable and responsible path is to set a careful course that can realize the promise while managing the peril.

This congressional hearing was party inspired by Bill Joy's cover story for *Wired* magazine, *"Why The Future Doesn't Need Us".* Bill Joy, cofounder of Sun Microsystems and principal developer of the Java programming language, has recently taken up a personal mission to warn us of the impending dangers from the emergence of self-replicating technologies in the fields of genetics, nanotechnology, and robotics, which he aggregates under the label "GNR." Although his warnings are not entirely new, they have attracted considerable attention because of Joy's credibility as one of our leading technologists. It is reminiscent of the attention that George Soros, the currency arbitrager and arch capitalist, received when he made vaguely critical comments about the excesses of unrestrained capitalism .

Joy's concerns include genetically altered designer pathogens, followed by self-replicating entities created through nanotechnology. And if we manage to survive these first two perils, we will encounter robots whose intelligence will rival and ultimately exceed our own. Such robots may make great assistants, but who's to say that we can count on them to remain reliably friendly to mere humans?

Although I am often cast as the technology optimist who counters Joy's pessimism, I do share his concerns regarding self-replicating technologies; indeed, I played a role in bringing these dangers to Bill's attention. In many of the dialogues and forums in which I have participated on this subject, I end up defending Joy's position with regard to the feasibility of these technologies and scenarios when they come under attack by commentators who I believe are being quite shortsighted in their skepticism. Even so, I do find fault with Joy's prescription: halting the advance of technology and the pursuit of knowledge in broad fields such as nanotechnology.

In his essay, Bill Joy eloquently described the plagues of centuries past and how new self-replicating technologies, such as mutant bioengineered pathogens and "nanobots" run amok, may bring back long-forgotten pestilence. Indeed these are real dangers. It is also the case, which Joy acknowledges, that it has been technological advances, such as antibiotics and improved sanitation, which have freed us from the prevalence of such plagues. Suffering in the world continues and demands our steadfast attention. Should we tell the millions of people afflicted with cancer and other devastating conditions that we are canceling the development of all bioengineered treatments because there is a risk that these same technologies may someday be used for malevolent purposes? Having asked the rhetorical question, I realize that there is a movement to do exactly that, but I think most people would agree that such broad-based relinquishment is not the answer.

The continued opportunity to alleviate human distress is one important motivation for continuing technological advancement. Also compelling are the already apparent economic gains I discussed above that will continue to hasten in the decades ahead. The continued acceleration of many intertwined technologies are roads paved with gold (I use the plural here because technology is clearly not a single path). In a competitive environment, it is an economic imperative to go down these roads. Relinquishing technological advancement would be economic suicide for individuals, companies, and nations.

3.1 The Relinquishment Issue

This brings us to the issue of relinquishment, which is Bill Joy's most controversial recommendation and personal commitment. I do feel that relinquishment at the right level is part of a responsible and constructive response to these genuine perils. The issue, however, is exactly this: at what level are we to relinquish technology?

Ted Kaczynski would have us renounce all of it. This, in my view, is neither desirable nor feasible, and the futility of such a position is only underscored by the senselessness of Kaczynski's deplorable tactics. There are other voices, less reckless than Kaczynski, who are nonetheless arguing for broad-based relinquishment of technology. Bill McKibben, the environmentalist who was one of the first to warn against global warming, takes the position that "environmentalists must now grapple squarely with the idea of a world that has enough wealth and enough technological capability, and should not pursue more." In my view, this position ignores the extensive suffering that remains in the human world, which we will be in a position to alleviate through continued technological progress.

Another level would be to forego certain fields—nanotechnology, for example—that might be regarded as too dangerous. But such sweeping strokes of relinquishment are equally untenable. As I pointed out above, nanotechnology is simply the inevitable end result of the persistent trend towards miniaturization that pervades all of technology. It is far from a single centralized effort, but is being pursued by a myriad of projects with many diverse goals.

One observer wrote:

> "A further reason why industrial society cannot be reformed…is that modern technology is a unified system in which all parts are dependent on one another. You can't get rid of the "bad" parts of technology and retain only the "good" parts. Take modern medicine, for example. Progress in medical science depends on progress in chemistry, physics, biology, computer science and other fields. Advanced medical treatments require expensive, high-tech equipment that can be made available only by a technologically progressive, economically rich society. Clearly you can't have much progress in medicine without the whole technological system and everything that goes with it."

The observer I am quoting is, again, Ted Kaczynski. Although one will properly resist Kaczynski as an authority, I believe he is correct on the deeply entangled nature of the benefits and risks. However, Kaczynski and I clearly part company on our overall assessment on the relative balance between the two. Bill Joy and I have dialogued on this issue both publicly and privately, and we both believe that technology will and should progress, and that we need to be actively concerned with the dark side. If Bill and I disagree, it's on the granularity of relinquishment that is both feasible and desirable.

Abandonment of broad areas of technology will only push them underground where development would continue unimpeded by ethics and regulation. In such a situation, it would be the less-stable, less-responsible practitioners (e.g., terrorists) who would have all the expertise.

I do think that relinquishment at the right level needs to be part of our ethical response to the dangers of 21st century technologies. One constructive example of this is the proposed ethical guideline by the Foresight Institute, founded by nanotechnology pioneer Eric Drexler, that nanotechnologists agree to relinquish the development of physical entities that can self-replicate in a natural environment. Another is a ban on self-replicating physical entities that contain their own codes for self-replication. In what nanotechnologist Ralph Merkle calls the "broadcast architecture," such entities would have to obtain such codes from a centralized secure server, which would guard against undesirable replication. I discuss these guidelines further below.

The broadcast architecture is impossible in the biological world, which represents at least one way in which nanotechnology can be made safer than biotechnology. In other ways, nanotech is potentially more dangerous because nanobots can be physically stronger than protein-based entities and more intelligent. It will eventually be possible to combine the two by having nanotechnology provide the codes within biological entities (replacing DNA), in which case biological entities can use the much safer broadcast architecture. I comment further on the strengths and weaknesses of the broadcast architecture below.

As responsible technologies, our ethics should include such "fine-grained" relinquishment, among other professional ethical guidelines. Other protections will need to include oversight by regulatory bodies, the development of technology-specific "immune" responses, as well as computer assisted surveillance by law enforcement organizations. Many people are not aware that our intelligence agencies already use advanced technologies such as automated word spotting to monitor a substantial flow of telephone conversations. As we go forward, balancing our cherished rights of privacy with our need to be protected from the malicious use of powerful 21st century technologies will be one of many profound challenges. This is one reason that such issues as an encryption "trap door" (in which law enforcement authorities would have access to otherwise secure information) and the FBI "Carnivore" email-snooping system have been controversial, although these controversies have abated since 9/11/2001.

As a test case, we can take a small measure of comfort from how we have dealt with one recent technological challenge. There exists today a new form of fully nonbiological self replicating entity that didn't exist just a few decades ago: the computer virus. When this form of destructive intruder first appeared, strong concerns were voiced that as they became more sophisticated, software pathogens had the potential to destroy the computer network medium they live in. Yet the "immune system" that has evolved in response to this challenge has been largely effective. Although destructive self-replicating software entities do cause damage from time to time, the injury is but a small fraction of the benefit we receive from the computers and communication links that harbor them. No one would suggest we do away with computers, local area networks, and the Internet because of software viruses.

One might counter that computer viruses do not have the lethal potential of biological viruses or of destructive nanotechnology. This is not always the case; we rely on software to monitor patients in critical care units, to fly and land airplanes, to guide intelligent weapons in our current campaign in Iraq, and other "mission-critical" tasks. To the extent that this is true, however, this observation only strengthens my argument. The fact that computer viruses are not usually deadly to humans only means that more people are willing to create and release them. It also means that

our response to the danger is that much less intense. Conversely, when it comes to self-replicating entities that are potentially lethal on a large scale, our response on all levels will be vastly more serious, as we have seen since 9/11.

I would describe our response to software pathogens as effective and successful. Although they remain (and always will remain) a concern, the danger remains at a nuisance level. Keep in mind that this success is in an industry in which there is no regulation, and no certification for practitioners. This largely unregulated industry is also enormously productive. One could argue that it has contributed more to our technological and economic progress than any other enterprise in human history. I discuss the issue of regulation further below.

Chapter 11

Current Status and Future Development of Structuring and Modeling Intelligent Appearing Motion

***

Thomas Alexander

FGAN - FKIE
Wachtberg-Werthoven
Germany

Stephen R. Ellis

NASA Ames Research Center
Moffett Field, California 94035-1000,
U.S.A.

"Our nature consists in motion, complete rest is death."
Blaise Pascal (1623-1662), French mathematician, physicist, and philosopher.

"Eppur si muove—Still, it moves."
Commonly attributed to Galileo Galilei (1564-1642), Tuscan astronomer, philosopher, and physicist.

The two topics covered by this symposium were intelligent appearing motion and Virtual Environments (VE). Both of these are broad research areas with enough content to fill large conferences. Their intersection has become important due to conceptual and technological advances enabling the introduction of intelligent appearing motion into Virtual Environments. This union brings new integration challenges and opportunities, some of which were examined at this symposium.

This chapter was inspired by the contributions of several of the conference participants, but is not a complete review of all presentations. It will hopefully serve as a basis for formulating a new approach to the understanding of motion within VE.

1. Virtual Environments

Virtual Environments (VE) and Virtual Reality (VR) are now often considered an innovative and natural interface for human-computer-interaction. But what is meant by VE and how does it differ from other human-computer-interfaces?

This section will briefly give some definitions and try to relate VE to other disciplines. The characteristics of VE will be specified, especially with regard to the impact of general and intelligent motion. Finally, a brief preview of future possible development is given.

1.1 What are Virtual Environments?

Virtual Environments (VE) provide new media for communication (Ellis, 1991). They subsume comprehensive technologies for presenting computer-generated scenes to human operators and enabling them to interact with them as if they were real (NATO HFM-021, 2001). VE often make use of multi-modality, including auditory and haptic in addition to visual stimuli. Inclusion of multiple modalities enhances the feeling of subjective presence ("feeling of being there") (Barfield & Furness, 1995; Stanney, 2004). The sense of subjective inclusion or immersion of the user in the computer-generated scene is generally stronger than it is with standard desktop IT-technology. Even when restricted to visual stimuli only, the immersion remains strong due to stereoscopic presentation and simulation of observers' real-time motion through a synthetic environment.

VE systems and displays vary with regard to the proportion of real versus virtual stimuli that they present (Kalawsky, 1993; Milgram et al., 1994). Immersing VE-displays, e.g., head-mounted displays (HMDs), totally exclude real world visual stimuli and present only synthetic stimuli. Other display systems described as Augmented Reality enable presentation of a mixed or augmented environment with virtual and real elements. In this case, synthetic parts may be spatially conformal and appear to be spatially integrated in the real environment (Azuma et al., 2001).

In addition to realistic presentation, interactivity is another basic characteristic of VE. It makes an active exploration and experience possible. This feature also supports the immersion into the virtual scene, especially for understanding of visually complex information. Obviously, interactivity is closely related to dynamics and motion, without these, interaction would not be possible.

However, technical limitations in presenting environmental stimuli and the awareness of being exposed to a VE rather than the real world may affect the interpretation of the environment and cause operators' behavior in a VE to differ from that in the real world. This is especially important in applications like simulator-based training, where training skills and knowledge is the main issue and incorrect training has to be avoided. In general, computer-based training systems are successful for training vehicle drivers, with everything within arm's reach being real and everything out-of-the-window being virtual. However, for training teams with multiple individual viewpoints these systems still need improvement.

FP Brooks' presentation on human motion in *VE for Team Training* illustrated the spectrum of possibilities for team training. For assessing the applicability of VE for training small teams he referred to system effectiveness and influential technological factors of a VE. In his experiments, he focused on observations of human motions and changes of various physiological measures of

effect in a compelling environment. Among others, the technological factors examined were field-of-view, method of travel within a VE, passive haptics, and latency. According to Brooks, a restricted field-of-view caused only limited behavioral differences. Early results assessing differing methods of travel within a VE showed that real walking in a VE closely mimicked reality, whereas an indirect "flying" technique produced motion pathes quite different from motion in the real world. Passive haptics supported the degree of immersion strongly.

1.2 Intelligent Agents and Avatars

Inclusion of virtual, intelligent organisms into the VE can make the system more useful and realistic. This feature is especially important for applications in education and training, when interaction and communication with other, synthetic entities is required. One big advantage of VE in comparison to live training is the more deterministic and totally reproducible setup of training scenarios, and the greater flexibility of the virtual system itself. With today's systems, such a high fidelity of the computer-generated scene can be achieved, that virtual simulation can approach and sometimes even replace live training to some extent.

When VE incorporates motion simulation of anthropomorphic entities or avatars, consistency between the visual appearance of the avatar and its motion becomes essential. A photorealistic rendering with only minimalist, abstract motion simulation is likely to appear incongruous. On the other hand, simple motion simulation for cartoon-like avatars may still seem realistic. Nevertheless, for applications with a high fidelity visual representation, like VE, a high fidelity of the behavior modeling is required.

Realistic motion of anthropomorphic elements within the VE, such as virtual humans, is critical because users of such systems are very sensitive to inaccuracies. From lifelong experience we have gained such detailed knowledge about gestures, facial expressions etc. that we notice even small errors and inconsistencies instantly. Furthermore, we often use motion as an indicator for inferring emotional states, intentions, and goals. Accordingly, slight inaccuracies in motion modeling might therefore easily lead to incorrect inferences about future actions and goals of virtual entities.

There are several approaches for implementing virtual humans with computer-generated behavior into VE. D. Thalmann presented several realistic virtual humans in his presentation on *Intelligent Virtual Humans Behavior.* According to him, the main problem areas were the level of AI within perception and motion control. Yet, only models for generalizing simple types of low-level motion existed, for instance, for walking or reaching. Future applications will require a single, more general model for low-level and high-level motion.

Most virtual human models were developed for computer graphics and related domains. These models look very realistic, especially when visualized as static pictures. But realistic animation has proven difficult and is often implemented off-line by manually programming motion sequences or by controlling the movements by motion recorded from a real actor. In both cases a simplified model is used to visualize the output first, and the final, photorealistic simulation is calculated in a considerably longer period afterwards. Real-time, dynamic photorealistic

rendering is still limited due to available computational resources and performance. Consequently, one still has to trade off photorealism with realistic motion behavior.

There is another discrepancy between modeling behavior on a low (movement) and high (behavioral, cognitive) level. Low-level modeling primarily focuses on single, goal-directed movements, like walking towards or reaching for a target, whereas high-level modeling refers to goals-seeking behavior and goal generation. The models simulating low-level motion behavior are frequently used for workplace design, games, or animation in the movies, and they enable realistic appearing motion for photorealistic human models. A more detailed description of the underlying modeling principles can be found in section 2.2.1 to 2.2.3.

On the other hand, high-level behavior models often lack high quality rendering and visualization. Instead, they model human performance and cognitive processes. Detailed information about them is given in section 2.2.4 and 2.2.5. These models are mainly used for the design of complex working processes including the human-in-the-loop and applications in simulator-based training, especially of vehicle drivers.

In his presentation on *Behavior of Synthetic Entities,* R. Kruk presented approaches for modeling decision-making and more complex behaviors for simulator-based training. It was based on real characteristics of the original system's performance and the decision-making processes actually used in practice. Additional knowledge about, for instance, terrain, culture etc. was included in the model. The behavioral models drove diverse (airborne and grounded) vehicle models within a simulation framework.

Which level or model to choose depends strongly on the specific application. In case, low-level motion is needed, a script may simply control behavior on a higher level to follow special instructional procedures and no high-level control is required. On the other hand, for training organizational processes and simulator-based training of situation management, higher-level modeling or realistic rendering is more important. In this case, low-level movement modeling may be simplified or ignored totally.

Because of the complexity of the whole field, only few multilevel models exist. But recent approaches in defining a common interface between models of different levels are leading towards a more comprehensive approach. They include a high-level behavioral model controlling a low-level movement model. This way the synthetic entity is able to act autonomously in the VE with realistic appearing movements.

The presentation of M. Sierhuis *About Subsumption Architectures in AI* addressed modeling individual behavior and connecting two different behavioral models. The NASA-developed software BRAHMS, a model for describing human behavior, was used to describe and analyze observations of a team in an isolated environment. Subsequently, the model controlled the motion of VE models of the virtual humans. Their behavior was modeled based on the pre-recorded behavioral patterns, and resulting effects of team interaction and actual operations were simulated.

Applications like this show that avatars and virtual agents now can be used for system design and work procedure design.

1.3 Adjustable VEs for Intelligent User Support

In addition to animation of the VE content, the capturing and processing of observer's ego-motion is another important aspect of motion in VE. VEs have been characterized before as highly interactive, enabling a natural human-system-interface. Since observer's motions often vary, so do interaction preferences. A simple example of this is handedness. Therefore, it will likely be necessary to customize the VE for each individual. This is especially important because of the close coupling between user actions and system's responses within typical VE-systems.

VE have the capability to enhance reality or to present a modified reality. This is not limited to Augmented Reality or Mixed Reality, but affects each VE application. For instance, navigating through a VE can provide both, global overview (exocentric viewpoint) and local situational awareness (egocentric viewpoint). In his presentation about *Optimizing Viewpoint Control in VE,* P. Milgram described alternative ways for providing spatial situation awareness while navigating and moving through a VE. In contrast to a pure exocentric, global frame of reference, or an egocentric frame of reference, he argued for a spring-damper-coupling between camera and motion combining some of the benefits of both, ego- and exocentric views. In this case the viewpoint was related to both, global overview and personal viewpoint.

But even when limiting the VE to a model of reality, it is not possible to incorporate all aspects of a real environment. Instead, they have to limit interaction capabilities and simulation to application-dependent features and to user-specific needs. For example, it would be senseless to visually simulate events that are beyond users' visual resolution. A future, intelligent VE system could take such limits into consideration and model only the relevant aspects of the environment.

One approach to this would be to include intelligent prediction algorithms for user interactions that take users' and systems' perceptual and motor limitations into account when adjusting parameters such as time-step size in motion dynamics calculations.

At a higher level, an intelligent system would have to include modules to learn the users' natural motions. It would use this knowledge for obviating the need to teach the user new interaction procedures. In this connection, the presentation of K.F. Kraiss on *Hierarchical Structured Nonintrusive Sign Language Recognition* provides an example of the benefits of such an approach since it is self-teaching and adaptive to new signs. By extending the gesture recognition of such systems to continuous gesturing, VE systems would become adjustable to user's natural gesture input. This makes interaction more natural and intuitive.

A further step towards an intelligent system would be to assess the user's preferences, state, and expectations. Out of this, consequences for interactivity and general scene design would have to be inferred. They have to be included in the general scene presentation to minimize additional modeling. The system has to monitor the users' actions continuously and adjust to their actual state. It has to adjust to shift resources dependent on new needs and requirements as a reaction.

The amount of monitoring and system's variability varies widely. Simple systems would include predictors for operator movement behavior to minimize latency effects. An application area, where this feature is crucial, is teleoperation. When using VE for teleoperation, there is high demand on realtime system response, especially when including haptic feedback. For obvious reasons, with an application like teleoperation in surgery this is of even more importance. F. Cardullo addressed the associated problems and proposes possible solutions in his presentation on *Telerobotic Surgery.* It was described how to overcome especially the transport delay problems by using an intelligent system to predict operator actions and reactions on the remote site.

With intelligent modules for the inclusion of further knowledge about system behavior the user performance can be enhanced for more complex teleoperation systems. A Grunwald showed in his presentation on *How Inverse Dynamics Make Users Smart* how supporting the user with adequate clues helps performing complex planning tasks during maneuvering space vehicles under dynamically, counter-intuitive conditions (e.g., the orbit environment with spatial movement restrictions and fuel usage constraints).

Applications like these show the need for including intelligence into the design of future VE systems. Today, VE systems make wide use of e.g., predictive Kalman filters to anticipate users' motion in order to overcome missing positional data and ensure a constant frame rate for rendering (Kalman, 1960). Future complex systems would relate to the computing power and shift computing processes in accordance to their user's preferences. By using operator behavior as input variables, they could serve as prototypes for intelligent VE.

2. Motion and Intelligent Appearing Motion

Motion, the second topic of the symposium, is essential for the perception and understanding of our environment. By simply looking at everyday reality, we can notice that it is never static. The environment itself either causes perceivable motion when objects themselves move, or we cause apparent environmental motion ourselves when we move through the environment.

Sometimes the observed motion appears intelligently driven. This section summarizes several aspects of the appearance and generation of intelligently driven motion. Additionally, different simulations of specifically humanoid motion within computer-graphics are discussed.

Physical motion in Virtual Environments

Physical motion of objects is a topic that has been studied extensively especially in mechanics (Newton, 1687; etc.). It is classically defined as *"…an act, process, or instance of changing place: movement; …an active or functioning state or condition"* (Merriam-Webster Collegiate Dictionary, 2003). Measuring the change of position of an object in a special amount of time quantifies motion leading to the computation of velocity as first derivative of place by time. Motion, however, need not be limited to a change of location, but could also metaphorically describe change of other attributes. Examples of such attributes could be light and color (e.g., due to lighting or shading), personal opinion, or even existence of the content itself in a VE (e.g., objects and sounds could appear or disappear).

Motion is always dependent on factors and physical principles defining constraints, features, and relations.

One example of this is *acceleration,* which describes changes in velocity during time. In a physical environment it is the quotient of force and mass. This simple relationship shows that there are generally two main factors affecting physical motion: Attributes of the static object itself (mass) and additional controlling factors (force). Both together cause a characteristic motion. By knowing all the internal attributes of an object, e.g., mass and other properties, and external factors, e.g., interacting forces, the motion of an object can be described and extrapolated into future. Even more complex motion of intelligent entities can be modeled when there is sufficient knowledge of internal characteristics and external factors available. In this case, internal characteristics are not limited to physical parameters but include psychological aspects as well, e.g., personality and motivation. External factors can either refer to low level of modeling, e.g., forces, obstacles, or to more complex levels, e.g., other persons interacting with the virtual entity. In either case, a complete understanding would make a comprehensive modeling possible.

Internal and external factors are manifold and numerous. As a matter of fact, the variety of possible motions and their combinations goes beyond the scope of modeling. A modeling of a definite motion is difficult or even impossible. Constraints or rules have to be specified in order to minimize the variability and find realistic solutions for motion modeling.

T. Sheridan addressed this issue in his presentation on *Constraint, Intelligence, and Control Hierarchy in Virtual Environments.* He pointed out that utility is the solution of two kinds of equations: on the one hand the objective (utility) function for the goodness of a solution and on the other hand the given constraint equations. This can be applied to language, music, body movements, graphic displays, computer programming and supervisory control, and VE. Constraints that apply in VE are mainly sensory range and resolution of the observer, observation point consistency in space and time, continuity of kinematics in space and time, cause and effect, mechanical impedance interaction with the observer/user, symbolic interchange, and etiquette. He concluded, that, if the many expected constraints are not adhered to, a VE does not appear real or even intelligent.

The reasoning behind the motion and the way it is executed are important for the sensation of realism of the behavior of the content of a VE and, consequently, of the VE itself.

But what makes motion appear intelligent? One approach to answer this question is to refer to the definition of intelligence in biology and to the contribution of behaviorists. W. Kosinski did so in his presentation on the *Pavlovian, Skinner, and Other Behaviorists' Contribution to AI.* Intelligence is defined as the ability for reasoning, imagination, insight and judgment. It requires three fundamental cognitive processes, which are: Abstraction, learning, and dealing with novelty. Behavioral intelligence was only noticed, when the observer saw how that behavior is adaptive. Therefore, intelligence was subjective or "in the eyes of the observer" (Brooks, 1991). Kosinski specifies AI approaches for modeling learning and stresses the importance of the inclusion of AI in complete systems design.

2.2 Simulation of Human Motion

Simulation of human movements has been a research topic in the fine arts, sports medicine and orthopaedics, workplace-design, safety, and, most recently, in computer graphics. A broad variety of computational models already exist for simulating realistic motion and complex behavior. However, most of the approaches used focus only on a single application. The baseline characteristics of such different simulation approaches were addressed in the following presentations at the meeting.

2.2.1 Motion Capture and Animation

In the following, actual approaches and methods of modeling human motion are briefly described. They involve a broad spectrum of diverse application areas like animation in computer graphics, workplace design, and basic research on human behavior.

Motion capture is a direct, straight-forward, and simple way to model motion: By attaching markers to moving objects and measuring the three-dimensional positions during movements, detailed, realistic trajectories are recorded. Reproducing these positions and mapping them on a computer-generated object gives the impression of a very high-fidelity motion simulation.

Nevertheless, this procedure is inadequate for modeling adaptive or intelligent behavior because there is basically no real motion model behind it. But it can serve to generate a database of primitive motions for a more sophisticated model, in which captured data is used to parameterize a motion model for animation of computer-generated objects.

Another way to animate an object is keyframe animation. In this case, captured position data of just a few *keyframes* serve as input. Positions in between are interpolated. With growing complexity of the interpolation algorithms, fewer keyframes are required for a realistic motion.

Such techniques are frequently used in games or movies, where motion follows simple rules or is even script-driven. The primary intention is not a valid model but a realistically appearing output. Most of such models are either special proprietary developments or modules of graphics and animation libraries.

Motions capture and animation are usually limited to movement modeling on a primitive level. They can seldom be applied for simulating complex behavior, especially in combination with goal selection and generation. Instead, they serve as a baseline vocabulary, which is a sort of a database of movements of behavior models of a higher level.

2.2.2 Biomechanic Models

Biomechanic models were originally developed for sports medicine and workplace design. They make use of laws of mechanics and apply them to the human body. The human body is considered as a complex mechanical system (including simple joints of different degrees-of-freedom with static connections between them). Motion is modeled, for instance, by applying

cost functions including forces, torques, load, and (potential and kinetic) energy. This procedure overlaps widely with algorithmic animation, so that a clear distinction between both is not possible.

The original focus of biomechanics was to calculate maximum loads of postures for workplace design and to optimize motion for better performance during energetic work. Biomechanics is often based on real motion capture data, but it generates a more sophisticated model from it. However, due to a large number of factors, the results may look less realistic than the originally captured data.

Like motion capture, biomechanics works primarily for motion modeling on a low level; for modeling behavior, other models have to be applied.

2.2.3 Digital Anthropometric Models

For an ergonomic workplace-design the inclusion of human dimensions has always been essential. At first, physical templates for representing small, medium and large persons were applied for this purpose. With growing importance of computer-aided design media, digital templates were developed and used for the design process. Whereas first digital models were just CAD-versions of the physical templates, not taking into account joint movement limits and applicable only for static applications, today's anthropometric models include complex algorithms for describing human body shape variability and basic human movement behavior. Behavior modeling is mostly based on biomechanic models, and refers only to simple, goal-directed movements like reaching for targets or gait modeling. For modeling of human behavior control and variability, no comprehensive approach exists.

The most common models are JACK (UGS, 2005), RAMSIS and Anthropos (Human Solutions, 2005), and SAFEWORK (Safework, 2005). Both models come with photorealistic rendering and movement modeling capabilities. However, their background and application area varies. The background of JACK is primarily computer graphics and animation. It is frequently used for computer-based training in VE. N. Badler, the model's developer, referred to this model in his presentation on *Meaningful Motion.* The main application area of RAMSIS is the automotive industry. With growing interest in VE in this field, integrating it into VE is currently being evaluated. In this application the designer controls the model by motion capture. For Safeway a similar approach is undertaken. It shows a general trend of inclusion of these detailed models into VEs for various purposes.

2.2.4 Performance Models

The original background of performance models is modeling human reliability and performance for resource and process planning and optimization. They consist of special modules for perception, cognition, and motor reaction. By combining these modules, conclusions about total human-machine-system reliability or performance for highly complex tasks can be inferred. Performance models are widely used to optimize interactive processes of different types.

MIDAS (Man-Machine Integration Design and Analysis Systems) is one of these models. It is an integrated suite of software components developed to aid in the design complex human-machine systems. The goal is to develop an engineering environment, which contains tools and models to assist in the conceptual phase of crewstation development, and to anticipate training requirements (MIDAS, 2005).

These models approach behavior simulation at a higher level. They refer primarily to goal selection and goal generation. With sufficient information and data, it is possible to use them for modeling human decision making during motion planning. In this case, a connection between a high-level performance model and a low-level animation model would make an intelligent model. However, this connection exists only in a very preliminary way in some models, and must still overcome associated technical problems.

2.2.5 Cognitive Models

Cognition is a very complex domain that places great demands on computational modeling. The general problem is that modeling cognition is somehow a modeling of the 2nd order; it is a kind of recursive modeling. This is because a cognitive model has to model a model of the real environment, which was generated by the cognitive system.

Most cognitive models are based on a formal analysis and inferred probabilistic models. They usually serve as a software-tool for solving problems and are based on fundamental knowledge from psychological research and on a variety of different case studies. Cognitive models simulate probabilities of special states and transitions between the states. For instance, working on different tasks sequentially defines a unique order of the tasks. Subjects might choose different orders based on their working strategy. The probabilities of the transitions from one stage to another can be used to model these strategies expressed in Bayesian terms or Hidden Markov Models.

The overall goal of cognitive models is to model human operator behavior with regards to his sensor, cognitive, and motor properties. They can be used for optimization of human-machine-systems at a very early design phase. By modeling operator behavior they can reduce the need for time-consuming experiments and minimize the design iterations for human-machine-system development.

One of these models is the *Atomic Components of Thought—Rational or ACT-R* (Anderson, 1996; CMU, 2005). It serves as a framework for modeling different tasks in a special programming language. This specific model can rely on general assumptions, which are provided by ACT-R or can be specified by the author. ACT-R has been used successfully for producing user models with human-computer interaction that can assess different computer interfaces, cognitive tutoring systems in education, cognitive agents that inhabit training environments and to interpret FMRI data. To some extent an application of such a model for goal-generation and goal-selection might be possible. In this case such a model could be applied to control the goal of a motion of a low-level movement model.

2.3 Perception of Motion and the Impact of Attribution

Despite the fact that motion is often considered to be purely an output process, it is closely related to perception and cognition. Motion is a closed loop, in which perception gives feedback about the effects of motions and movements, resulting again into adjustments for further actions. In a general sense, perception gives us an image and understanding of the environment, and motion is used to respond with appropriate actions. Both areas are closely coupled to each other so that separating them would make it difficult to understand and simulate either of them.

D. Wolpert considered sensory and motor uncertainties forming fundamental constraints on human sensorimotor control. In his presentation on *Uncertainty and Prediction in Sensorimotor Control* he presented the effect of several sensory and motor factors, e.g., noise, on motion. Factors with a major effect on the sensorimotoric system were identified. For instance, Wolpert found 30-70% sensory underestimations of self-produced force, which might be responsible for force escalation in an interpersonal exchange of blows in a kind of a two-person shoving match.

The study of the perception of motion is crucial for generating intelligent appearing motion in VE. For an overall sensation of motion it seems not to be important how exactly low-level motion is modeled, but whether an overall cause or explanation for it can be inferred. In this case the individual disposition, motivation and knowledge of the observer allows him to infer this explanation. He “attributes” to events and these attributions help him to understand even complex situations. The attribution theory, initiated by Fritz Heider in 1958, focuses on the study of the baseline concepts behind this. Causality and intention are two attributions that contribute especially to the perception of motion.

The experiments of Michotte (1946) on causality demonstrate that causal interaction can be perceived with exceedingly simple visual displays. For example, one circle was shown moving from the left to the middle of a display, and a second one moved afterwards from the middle to the right. For different temporal lags or spatial gaps, subjects reported either a continuous motion with one ball hitting the other (causality) or two separate movements (no causality). In his experiments, Michotte found 50 ms to be the threshold value of perceiving causality. With regard to spatial differences, speed was determined to be the all-important factor: With larger speeds the gap could be larger without constraining causality. Notice photorealism was not required to perceive causality.

Metzger (1934) describes another example showing the impact of multimodality on perception of causality. Two identical visual targets moving across each other can be perceived either to bounce off or to stream through each other. By introducing a brief sound at the coincidence most subjects report sensing a bounce. The reason for this sensation and perception is still a topic of actual research. Newtson (1976) supposed that observers seem to unconsciously segment behaviors into actions. Furthermore, expectation leads to inferring actions, preparing and anticipating actions before they take place. If the perceived action is consistent with the expectation, causality is inferred.

In further experiments of Michotte, subjects reported an object either “chasing” and “following” the other, or “guiding” it, dependent on the sizes of the proceeding and succeeding object. These sensations arose due to cognitive attribution during the experience of exocentric motion. The setup of these experiments is illustrated in Figure 1. Notice again the low realism of the setup.

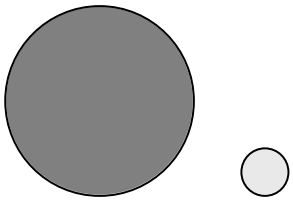


Figure 1. Setup of one of the experiments on causality by Michotte (1946). Subjects reported the small ball guiding the large ball.

In their experiments on the effect of attribution on motion perception, Heider & Simmel (1944) found that even for very abstract motions with geometric shapes human observers tend to build a story around it. In their experiments they used a setup as shown in Figure 2, which consists of an abstract set of two triangles, a circle, and a square that moved around. Subjects watching the scene created a sort of love story around with the circle falling in love with one of the triangles. Haider and Simmel explained these interpretations by the subject’s personal attribution.

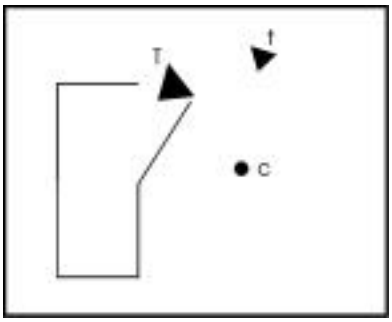


Figure 2. Setup of Heider and Simmel (1944). Subjects reported a love-story of the circle falling in love with one of the triangles and being chased by the other triangle.

Each of these experiments shows that, even without photorealistic appearance and high level, detailed simulation, motion can still appear realistic and intelligent. Such findings contradict an understanding of motion as a purely physical concept. Possible explanations are based on individual perceptual characteristics like attribution. Attribution, as cited before, helps the observer to find an explanation for events and answer the question “why.” It results in perceiving basic concepts, for instance, causal links between events.

Causality can be inferred readily if there is a spatial and/or temporal close relationship between two events. *Causal motion* would be an initial level of an additional structure subjectively needed for understanding the appearance of motion. It refers to the relationship of events or movements to each other. Causality is, of course, primarily based on mechanical laws, and can therefore be modeled. However, it is often intuitively inferred, based on experience and knowledge. Because of this attribution, perception of causality is not limited to perception but includes cognition also.

While causality refers to a relationship between single events without more complex interpretation, the theory of attribution hints at higher levels. For a set of events, a goal or an intention is often inferred. This goal is based on the observer's subjective thoughts and feelings and might sometimes be wrong. This level would be the level of *intentional movement.* It refers to the observer's perception and attribution of qualities to the motions.

In his presentation on *The Illusion of Sentience in Virtual Environments,* M. Slater focused on this aspect. In his experiments he found that subjects, especially with anxieties, reacted to virtual humans and interpreted their behaviors as if they were real. This reaction was especially surprising because the simulation of humans was relatively simple and subjects were aware totally of the situation. Especially in setups with anxiety in social situations, the subjects perceived the virtual humans as characters, not as synthetic entities.

These results show that subjective factors like attribution and sensation have a significant impact on the appearance of intelligent motion. Even though there are discrepancies and inaccuracies in geometric or behavioral modeling, a situation might still appear realistic and motion might still appear intelligent because of these perceptual and cognitive factors. Consequently, intelligent motion does not only have an objective dimension, which relates to the scene content and virtual entities, but also a subjective dimension, which relates to the observer. This fact requires additional efforts to specify rules and constraints for modeling intelligent appearing motion.

2.4 Personal Traits and Motion

Causation is not the only subjective attribute that observation of motion may evoke. More complex cognitive attributes may be elicited by the semantic content of a complex scene. For understanding complex scenes different levels of perception and knowledge domains are required. If knowledge from one level is incomplete, knowledge from others may be recruited. An example may be found in the behavior of the entities of Heider's experiments referred to in the previous section, where behavioral explanation spontaneously extended beyond a purely mechanical behavior . In this case social knowledge was recruited to interpret the scene.

Inferring and understanding the semantic content of a situation in a VE can involve hierarchical classifications of objects, their motion, and their behaviors. At the first level, the *primitive categorization,* objects are categorized into animate or inanimate objects based on their behavior. The behavior of animate objects incorporates some sort of intelligence, while inanimate objects show a simple, mechanical behavior only. For animate objects, a second classification may follow. It refers to *primitive psychology,* which includes essential needs of living beings like hunger, thirst, and sleep as the motivation for a special behavior. The third level of categorization refers to *folk-psychology* or naïve psychology. It describes a common understanding of mental

states based on our everyday ascriptions and also includes complex concepts like belief, desire, fear and hope (Fodor, 1987; Goldman, 1993) as motivation. A higher level would introduce *personality traits* to the virtual entity, like extroversion, agreeableness, conscientiousness, emotional stability, and intellect (Wiggins, 1996). Adding these characteristics to behavioral modeling can make the entities' physical actions more transparent to observers and provides the virtual entity with a unique character and behavior.

N. Badler referred to this aspect in his presentation on *Meaningful Motion.* By adding an additional controlling module to a digital human model it was possible to adjust the synthesis of a motion so that specific character attitudes may be observed. In this connection, personality traits and other characteristics are understood as a quality of the motion. The motion itself then still remained goal-directed.

For interaction beyond simple mechanical, *social rules* have to be considered. They include, e.g., occupancy roles, family roles, and stereotypes. A final level for reaching the highest anthropomorphism within a virtual scene would be the inclusion of *high social and political elements* like moral judgment and compassion into the virtual storytelling.

Each of these levels helps an observer understand motion in a virtual scene and behavior of virtual entities. Because of attribution the understanding of the semantic content of the scene is based on the individual knowledge and therefore may differ between observers.

By mixing and applying knowledge from various domains, e.g., psychology and drama, instead of limiting virtual scene generation to pure mechanics, motion and behavior of synthetic entities become much more realistic. This is even true when a single movement or entity appears less realistic, like, e.g., in cartoons or in an abstract setup.

However, inaccuracies or inconsistencies in behavior modeling impair the sensation of naturalistic motion strongly. It has to be pointed out that human observers are very sensitive in detecting deficits due to the extensive knowledge of the relationship between behavior and personality characteristics we have gained during our lifetime.

Relevant knowledge of these basic rules can be found in some surprising fields: Narration and drama. W. L. Johnson's presentation on *Expression, Intention, and Context in Animated Agents* focused on embodied conversational agents and on realistic virtual storytelling. He derived general recommendations for convincing observers that they are observing intelligent agents. The recommendations were originally taken from opera and drama.

While scientific approaches for behavior modeling are often very detailed and complicated, other, more straightforward approaches seem to give good and realistic results. They are often used within the game industry. J. Buchanan referred to possible synergies and differences between these approaches in his presentation on *Video Games: Where Do We Go Now?* The presentation stressed possible applications in edutainment, i.e., education and entertainment, and especially the importance of an effective storytelling for a VE.

3. Comparative Analysis of Descriptions of Intelligent Motion

There are several different ways to classify the large variety of possible motion. A simple, first approach is to differentiate between a pure physical motion on the one hand and intelligently controlled motion on the other. Physical motion would subsume simple movements of dumb, passive objects and low-level movements of animated, active objects. Intelligent motion would focus more on the control behind the movement.

In his opening remarks, S. Ellis proposed such a cybernetic approach for structuring the complex of motion. In this approach, geometry of static objects serves as the basis and includes information about static positions, and restrictions of movable objects or joints. On the next higher level, dynamics are introduced, which refers to changes through time including factors, such as forces and torques. So far each level is based on physical behavior and physical laws. They define the visible output of motion in the VE. With the cybernetic level, control and more complex behavior are introduced. Higher levels relate to goal selection (teleotic level) and goal synthesis (geneotic level).

In this case, behavior is considered a subset of subordinate levels, including goal generation, goal selection, comparison of goal and state, and single movements itself. The approach enables detailed modeling and simulation. A model based on this approach is highly adaptable and would not depend on a specific environment. It would comprise simple movements as well as complex intelligent behaviors.

Current motion models are not able to include this whole spectrum. They are specialized either on modeling lower, i.e., movements, or higher, i.e., behavioral, level motion. Most of them are limited to special applications only and need major transfer modifications for others. But ongoing activities in merging different models show that first approaches for more complex models are coming up.

Nonetheless, the general shortcoming of a purely mechanic/cybernetic approach is that it does not consider the perception of motion and the elicitation of motion attributes. As mentioned before, interrelationships between events (e.g., causality) are often inferred as part of the understanding of more complex behaviors. In these cases personal knowledge is used to fill in the gap due to missing information.

Even simulating simple relationships like causality associated with simulated physical impact gets complicated without considering the perceptual and cognitive processes of the observer. For simulating social relations, it gets much more complicated and complex.

Instead, an alternative description of the relationship between events may be more suitable. This would focus more on the relation between events rather than simulating single, isolated events.

4. An Alternative, Linguistic Approach

Relationships and links between different motions are essential for motion perception and understanding complex intelligent behavior. Physical motion is just one aspect of motion, but causal and intentional movements create the perception and illusion of intelligence within motion. Therefore, the original cybernetic approach for structuring the variability of motion may be extended into another dimension, taking into account explicitly the links and relationships between events and motions that support the concatenation of motion segments.

One possibility would be to include findings in linguistics and semiotics. In this case, motion is considered as a media for communication, instead of being limited to purely goal-directed movements. This extension is important, because VE was previously defined as a new media for communication. A *linguistic approach,* as shown in Figure 3, focuses on a realistic communication between the (virtual) human with the (real) observer/user, on relations between movements, movements and intention, and movements and acting person. In this approach, motion is referred to as a concept of hierarchically constituted entities and patterns, like words and sentences in language. Motion patterns are stored and simply retrieved when needed following syntactic rules.

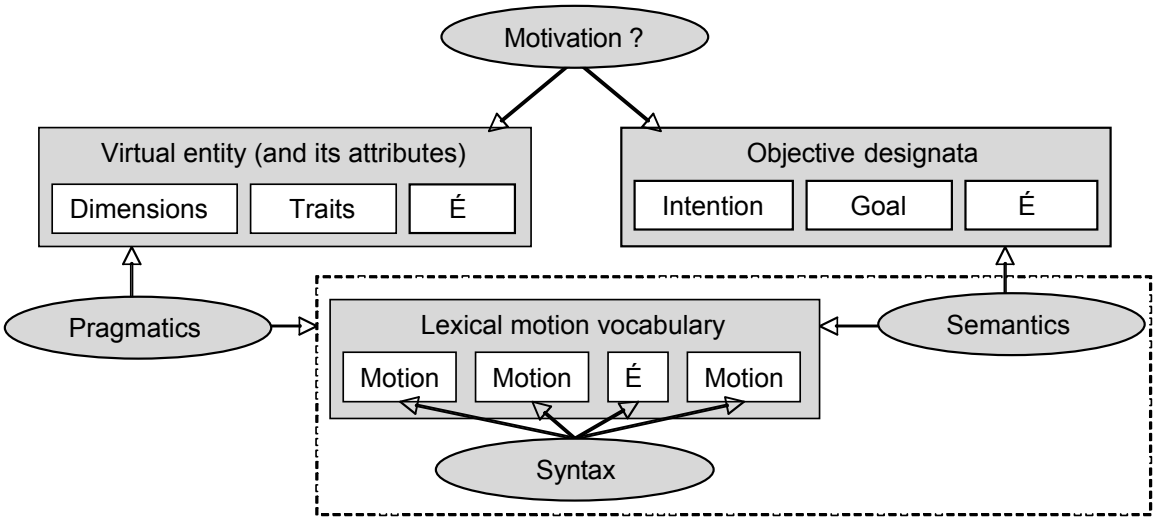


Figure 3. Alternative analytic structure of motion.

At a *lexical* level, it includes basic movements and their relation to static and geometric attributes of the moved entity. Basic kinematics calculated from these attributes are simply retrieved from a movement memory. Interrelations between movements of different parts of the objects, e.g., joint movements, are also described within the lexical level. As a result, the lexical level consists of a small set of simple movements, like words.

The combination of single lexical units to more complex motions happens at the next higher level, which is the *syntactic* level. At this level the relationship between single movements is specified, based on traits of the acting virtual entity. Several single units are combined to form a more complex motion. Transitions between single movements have to be determined to assure continuity. In this level, basic concepts like causality and rules for causality can be specified.

At the *semantic* level, which relates to the designata, movement patterns are correlated to the meaning or overall goal of the motion. The relation between them and the goal of the motion is specified. By this, an observer is able to infer intention and understand even more complex situations.

Intelligence and autonomy are extended on the *conceptual* level. At this level, new goals and concepts for motions are generated. The conceptual level models behavior on a higher level. It has to relate to external factors, like environmental stimuli, as well as internal factors, like motivation and traits of the acting entity.

The relationship between motion and acting person has not been included in this structure. It is handled on the *pragmatic* level, which stands above the other levels. It refers to the relationship between the user and the communications medium.

This approach for structuring motion is based more on the relationship between elements of motion and the overall communicative purpose of the system. It considers motion as a further media to enhance communication between the system and the user. Like storytelling in the movies, drama, or in training concepts, it stresses interactivity and the compositional aspect of complex movement of the VE. By stressing the communication character of motion, a limitation to the communicative-relevant parameters of motion is achieved, whereas simulation of events not relevant to communication can be ignored.

5. Relevance and Future Applications of the Topic

The general trend of motion simulation within VE is for the inclusion of a consistent intelligent behavior for virtual entities, especially virtual humans, on different behavioral levels. Initially, simulation was limited to simple mechanical motion models, e.g., only very simple, robot-like motions and primitive behaviors were modeled. With growing computational resources, simulations include increasingly complex motions today. But still models are very specialized and limited for a single application area only. Only a few of them include more than one level of motion and behavioral modeling. They enable visible realistic movements of the virtual entity as well as selection and generation of underlying goals.

For modeling comprehensive virtual humans in VE only a few approaches exist. This situation is likely to improve in the future because of the development in several application fields.

Today's most common applications of intelligent motion are Virtual Environment systems for education and training. There is a growing demand on realistic training scenarios, which allow training of social skills under different environmental and cultural circumstances. Yet, this is almost exclusively done in physical training areas where the "inhabitants" of a scene are paid

actors and only limited special settings may be trained. This situation is obviously very time- and cost-intensive. Transferring these scenarios into a VE has the big advantage of higher reproducibility and total control of environment variables.

Another application of intelligent motion is related to teleoperation and telepresence. By finding a more efficient way to describe motion pattern with a minimum set of parameters, it would be possible to reduce required transferred information to this set. Ideally, the (intelligent) motion capture system would be able to identify and parameterize motion from a video stream. The consequent parameters would be transmitted to the remote system and control the motion of the remote entity. Especially, when there is only limited bandwidth available, data compression without information loss is required. Such a teleoperation system separates motion information from static (geometric) information, and reduces the amount of data drastically. The transmitted stream would look like a screenplay, referring to objects and their motion.

A more long-term application area is e-commerce and growing use of the World Wide Web. The reason for using anthropomorphic agents or "soft-bots" here is to make boring commerce look more real and to minimize impediments for potential customers. The agent represents an assistant, who aids the visitors of a webpage by helping them find what they are seeking on the page. Speech-understanding software may also be integrated, so that natural language can serve as user input. The agent's reactions include gesture, facial expression and, of course, verbal output. Today's agents are relatively simple and consist of different pictures of synthetic characters, but the development towards virtual, three-dimensional avatars is expected. By applying intelligent motion and "friendly" traits of personality to the virtual human, a friendly and educated assistant may become possible.

A further step would lead to the application of such a virtual human as an intelligent user-interface for a complex system. Complex systems tend to transfer their complexity to the user-interface (UI). For a correct mental model of system status a large amount of user's system knowledge is necessary. By using an anthropomorphic UI, it might be possible to make use of basic inherent social knowledge to enable information transfer in a very realistic and intuitive way.

References

Anderson, J. R. (1996): ACT: A simple theory of complex cognition. American Psychologist, vol. 51, pp. 355-365.

Azuma, R., Baillot, Y., Behringer, R., Feiner, Julier, S. S., and MacIntyre, B. (2001): Recent Advances in Augmented Reality. IEEE Computer Graphics, vol. 21, no. 6, pp. 34-47.

Barfield, W. and Furness, T. A. (1995): Virtual Environments and Advanced Interface Design. New York: Oxford University Press.

CMU (2005): About ACT-R. WWW-document obtained Jan. 05, 2005 from http://act-r.psy.cmu.edu/about/

Ellis, S. R. (1991): Nature and Origins of Virtual Environments: A bibliographical essay. Computing Systems in Engineering, vol. 2, no. 4, pp. 321-341.

Fodor, J. (1987): Psychosemantics: The Problem of Meaning in the Philosophy of Mind. Cambridge, Mass.: MIT Press.
Goldman, A. I. (1993): The Psychology of Folk Psychology. Behavioral and Brain Sciences, vol. 16: pp. 15-28.
Heider, F., and Simmel, M. (1944): An experimental study of apparent behavior. American Journal of Psychology, vol. 13.
Human Solutions (2005): Human Solutions GmbH. WWW-document obtained Jan 5, 2005 from http://www.human-solutions.de/
Kalawsky, R. S. (1993): The science of virtual reality and virtual environments. Wokingham, UK: Addison-Wesley.
Kalman, R. E. (1960). A New Approach to Linear Filtering and Prediction Problems. Transaction of the ASME—Journal of Basic Engineering, 82D, pp. 35-45.
Merriam-Webster (2003): Merriam-Webster's Collegiate Dictionary, 11th Edition. New York: Merriam-Webster.
Metzger, W. (1934): Beobachtungen über phänomenale Identität. Psychologische Forschung, vol. 19, pp. 1-60.
Michotte, A. E. (1946/1963): The Perception of Causality. New York: Basic Books.
MIDAS (2005): MIDAS Homepage. WWW-document obtained Jan. 5, 2005 from http://www-midas.arc.nasa.gov/
Milgram, P., Takemura, H., Utsumi, A., and Kishino, F. (1994): Augmented reality: A class of displays on the reality-virtuality continuum. SPIE, vol. 2351, Telemanipulator and Telepresence Technologies.
NATO HFM-021 (2001): What Is Essential for Virtual Reality Systems to Meet Military Human Performance Goals? Neuilly-sur-Seine: NATO RTA.
Newton, I. (1687): Philosophiae Naturalis Principia Mathematica. Facsimile version by Koyré, A., and Cohen, I. B. (1972), Cambridge, UK: Cambridge University Press.
Newtson, D. (1976): Foundations of attribution: the perception of ongoing behavior. In John H. Harvey, William J. Ickes, and Robert F. Kidd, editors, New Directions in Attribution Research: vol. 1, pp. 223-247, Lawrence Erlbaum Associates.
Safework (2005): Human Modeling Technology. WWW-document obtained Jan. 5, 2005, from http://www.safework.com/
Stanney, K. M. (2004): Handbook of virtual environments. Mahwah, NJ: Lawrence Erlbaum Associates.
UGS (2005): E-factory JACK. WWW-document obtained Jan. 05, 2005 from http://www.ugs.com/products/efactory/jack/
Wiggens, J. S. (1996): The Five-Factor Model of Personality. New York: Guilford Press.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
15/06/2007	Conference Proceedings	15/09/2003–17/09/2003

4. TITLE AND SUBTITLE
Intelligent Motion and Interaction Within Virtual Environments

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Stephen R. Ellis,[1] Mel Slater,[2] and Thomas Alexander[3]

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER
21-131-20-30-00

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
[1]Ames Research Center, Moffett Field, CA 94035-1000
[2]University College, London, U.K.
[3]FGAN-FKIE, Wachtberg-Werthoven, Germany

8. PERFORMING ORGANIZATION REPORT NUMBER
A-070003
TH-061

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
National Aeronautics and Space Administration
Washington, D.C., 20546-0001

10. SPONSORING/MONITOR'S ACRONYM(S)
NASA

11. SPONSORING/MONITORING REPORT NUMBER
NASA/CP–2007-213468

12. DISTRIBUTION/AVAILABILITY STATEMENT
Unclassified—Unlimited Distribution: Nonstandard
Subject Category: 53
Availability: NASA CASI (301) 621-0390

13. SUPPLEMENTARY NOTES
Point of Contact: Stephen R. Ellis, Ames Research Center, MS 262-2, Moffett Field, CA 94035-1000, (650) 604-6147, sellis@mail.arc.nasa.gov

14. ABSTRACT
What makes virtual actors and objects in virtual environments seem real? How can the illusion of their reality be supported? What sorts of training or user-interface applications benefit from realistic user-environment interactions? These are some of the central questions that designers of virtual environments face. To be sure simulation realism is not necessarily the major, or even a required goal, of a virtual environment intended to communicate specific information. But for some applications in entertainment, marketing, or aspects of vehicle simulation training, realism is essential. The following chapters will examine how a sense of truly interacting with dynamic, intelligent agents may arise in users of virtual environments. These chapters are based on presentations at the London conference on Intelligent Motion and Interaction within a Virtual Environments which was held at University College, London, U.K., 15–17 September 2003.

15. SUBJECT TERMS
Virtual environments, Artificial intelligence, Human factors, User interface, Virtual reality

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Stephen R. Ellis
Unclassified	Unclassified	Unclassified	Unclassified	191	19b. TELEPHONE *(Include area code)* (650) 604-6147